# Substituting quantum entanglement for communication

Richard Cleve[1,*] and Harry Buhrman[2,†]
[1]*Department of Computer Science, University of Calgary, Calgary, Alberta, Canada T2N 1N4*
[2]*CWI, P.O. Box 94070, 1090 GB Amsterdam, The Netherlands*

(Received 30 January 1997; revised manuscript received 16 April 1997)

We show that quantum entanglement can be used as a substitute for communication when the goal is to compute a function whose input data are distributed among remote parties. Specifically, we show that, for a particular function among three parties (each of which possesses part of the function's input), a prior quantum entanglement enables one of them to learn the value of the function with only two bits of communication occurring among the parties, whereas, without quantum entanglement, three bits of communication are necessary. This result contrasts the well-known fact that quantum entanglement cannot be used to simulate communication among remote parties. [S1050-2947(97)02908-9]

PACS number(s): 03.65.Bz, 89.70.+c

If a set of entangled particles are individually measured, the resulting outcomes can exhibit ''nonlocal'' effects [1–5]. These are effects that, from the perspective of ''classical'' physics, cannot occur unless ''instantaneous communications'' occur among the particles, which convey information about each particle's measurement to the other particles.

On the other hand, no communication actually occurs among the entangled particles when they are measured. To phrase this in operational terms, entangled particles cannot be used to simulate communication. For example, if two physically separated parties, Alice and Bob, initially possess particles whose quantum states are entangled and then Bob obtains a bit of information $x$, there is no operation that Bob can apply to his particles that will have the effect of conveying $x$ to Alice when she performs measurements on her particles. Moreover, entanglement cannot even be used to *compress* information: for Bob to convey $n$ bits (with arbitrary values) to Alice, he must send $n$ bits—sending $n-1$bits will not suffice. Also, similar results apply to communications involving more than two parties.

Consider the following related but different scenario. Alice obtains an $n$-bit string $x$, and Bob obtains an $n$-bit string $y$ and the goal is for Alice to determine $f(x,y)$, for some function $f:\{0,1\}^n\times\{0,1\}^n\rightarrow\{0,1\}$,with as little communication between Alice and Bob as possible. This can always be accomplished by Bob sending his $n$ bits to Alice, but fewer bits may suffice. For example, for the function

$$f(x,y)=x_1+\cdots+x_n+y_1+\cdots+y_n \tag{1}$$

(where + means addition modulo two), it suffices for Bob to send a single bit (namely, $y_1+\cdots+y_n$) to Alice. On the other hand, for other functions, such as the *inner product* (in modulo two arithmetic)

$$f(x,y)=x_1\cdot y_1+\cdots+x_n\cdot y_n\,, \tag{2}$$

$n$ bits of communication turn out to be necessary (see Ref. [6] for a proof of this). Thus, even though the goal is for Alice to acquire a single bit of information, this bit depends on the $2n$ bits distributed among Alice and Bob in such a way that they must exchange $n$ bits between them in order for Alice to determine this bit. For a function $f:\{0,1\}^n\times\{0,1\}^n\rightarrow\{0,1\}$, the minimum number of bits that must be communicated between Alice and Bob in order for Alice to determine $f(x,y)$ is called the *communication complexity* of $f$. Several aspects of communication complexity are surveyed in Ref. [6].

The question that we consider is whether or not a prior quantum entanglement can reduce communication complexity. For example, if Alice and Bob initially possess entangled particles, can they compute some functions using less communication than would be required without the entangled particles? Although we do not presently know the answer for this two-party scenario, we exhibit an analogous three-party scenario where entanglement *does* reduce communication complexity. The function is based on Mermin's version [5] of ''Bell nonlocality without probabilities.''

Consider the following three-party scenario. Alice, Bob, and Carol receive $n$-bit strings $x$, $y$, and $z$, respectively, which are subject to the condition that

$$x+y+z=\mathbf{1}, \tag{3}$$

where + is applied bitwise (modulo two) and

$$\mathbf{1}=\overbrace{11\ldots1}^{n}.$$

The goal is for Alice to determine the value of

$$f(x,y,z)=x_1\cdot y_1\cdot z_1+\cdots+x_n\cdot y_n\cdot z_n\,. \tag{4}$$

An alternative way of expressing this problem is to impose no restriction on the inputs, $x$, $y$, $z$, and to extend $f$ to a *relation* such that on the points where Eq. (3) is violated, both 0 and 1 are acceptable outputs. Clearly, this problem has the same communication complexity as the original one. We show that, for the cases where $n\geqslant3$: (i) without a prior entanglement, three bits of communication are *necessary* for Alice to determine $f(x,y,z)$; and (ii) with a certain prior

*Electronic address: cleve@cpsc.ucalgary.ca
†Electronic address: buhrman@cwi.nl

entanglement, two bits of communication are *sufficient* for Alice to determine $f(x,y,z)$. Thus, even though entanglement cannot be used to simulate communication, it can nevertheless act as a *substitute* for communication when the goal is to compute a function with distributed data. We also show that the lower bound of three in the case of no entanglement cannot be improved. This is done by exhibiting a three-bit protocol.

Recently, Grover [7] has independently demonstrated that quantum entanglement can reduce communication complexity in a different context.

## A TWO-BIT QUANTUM PROTOCOL

We now show that if $A$(lice), $B$(ob), and $C$(arol) initially share a certain entanglement of qubits then there is a protocol in which $B$ and $C$ each send a single bit to $A$, which enables $A$ to determine $f(x,y,z)$ [as defined by Eqs. (3) and (4)].

The entanglement involves $3n$ qubits, with each party having $n$ of them. Call the $n$ qubits that party $p\in\{A,B,C\}$ starts with $\mathbf{q}_1^p,\ldots,\mathbf{q}_n^p$. For each $i\in\{1,\ldots,n\}$, let the triple $\mathbf{q}_i^A\mathbf{q}_i^B\mathbf{q}_i^C$ be in state

$$\tfrac{1}{2}(|001\rangle+|010\rangle+|100\rangle-|111\rangle). \tag{5}$$

(This is equivalent to the state examined in Ref. [5] but in an alternate basis.) For convenience, in this section, we write $x^A$, $x^B$, and $x^C$ for the inputs of $A$, $B$, and $C$, instead of $x$, $y$, and $z$, respectively. Thus, each party $p\in\{A,B,C\}$ has qubits $\mathbf{q}_1^p,\ldots,\mathbf{q}_n^p$ and input string $x^p=x_1^p\cdots x_n^p$, and the goal is for party $A$ to determine the value of $f(x^A,x^B,x^C)$.

The protocol begins by each party $p\in\{A,B,C\}$ performing the following operations and measurements on his qubits in order to obtain a bit $s^p$:

```
for each $i\in\{1,\ldots,n\}$ do
    if $x_i^p=0$ then apply $H$ to $q_i^p$
    measure $q_i^p$ yielding bit $s_i^p$
$s^p\leftarrow s_1^p+\cdots+s_n^p$.
```

In the above, $H$ is the Hadamard transform, that maps $|0\rangle$ to $(1/\sqrt{2})(|0\rangle+|1\rangle)$ and $|1\rangle$ to $(1/\sqrt{2})(|0\rangle-|1\rangle)$ (and we recall that $+$ is in modulo two arithmetic). Also, all measurements are in the standard basis consisting of $|0\rangle$ and $|1\rangle$. Next, $B$ and $C$ send bits $s^B$ and $s^C$, respectively to $A$, who outputs the value of $s^A+s^B+s^C$.

This protocol works if and only if, for all $x^A,x^B,x^C\in\{0,1\}^n$ such that $x^A+x^B+x^C=\mathbf{1}$, the bits $s^A,s^B,s^C$satisfy

$$s^A+s^B+s^C=f(x^A,x^B,x^C). \tag{6}$$

The proof that Eq. (6) holds is based on the following lemma, which is equivalent to the result in [5], though expressed in a different language.

*Lemma 1*: For all $i\in\{1,\ldots,n\}$,

$$s_i^A+s_i^B+s_i^C=x_i^A\cdot x_i^B\cdot x_i^C. \tag{7}$$

*Proof*: By Eq. (3), $x_i^Ax_i^Bx_i^C\in\{001,010,100,111\}$.

First, consider the case where $x_i^Ax_i^Bx_i^C=111$. In this case, no $H$ transformation is applied to any of $\mathbf{q}_i^A,\mathbf{q}_i^B,\mathbf{q}_i^C$.Therefore, $\mathbf{q}_i^A,\mathbf{q}_i^B,\mathbf{q}_i^C$is measured in state (5), which implies that $s_i^A+s_i^B+s_i^C=1=x_i^A\cdot x_i^B\cdot x_i^C$.

Next, in the case where $x_i^Ax_i^Bx_i^C=001$, an $H$ transformation is applied to $\mathbf{q}_i^A$ and to $\mathbf{q}_i^B$ but not to $\mathbf{q}_i^C$. Therefore, $\mathbf{q}_i^A\mathbf{q}_i^B\mathbf{q}_i^C$ is measured in state

$$H\otimes H\otimes I(\tfrac{1}{2}(|001\rangle+|010\rangle+|100\rangle-|111\rangle))$$
$$=\tfrac{1}{2}(|011\rangle+|101\rangle+|000\rangle-|110\rangle) \tag{8}$$

so $s_i^A+s_i^B+s_i^C=0=x_i^A\cdot x_i^B\cdot x_i^C$. The cases where $x_i^Ax_i^Bx_i^C=010$ and 100 are similar by the symmetry of state (5). □

Now, it follows that

$$\begin{aligned} s^A+s^Bs^C &= \left(\sum_{i=1}^{n} s_i^A\right)+\left(\sum_{i=1}^{n} s_i^B\right)+\left(\sum_{i=1}^{n} s_i^C\right)\\ &=\sum_{i=1}^{n}(s_i^A+s_i^B+s_i^C)\\ &=\sum_{i=1}^{n}x_i^A\cdot x_i^B\cdot x_i^C\\ &=f(x^A,x^B,x^C). \end{aligned} \tag{9}$$

## NO TWO-BIT CLASSICAL PROTOCOL EXISTS

We show that, in the case where $n=3$, without the use of entangled particles, two bits of communication among Alice, Bob, and Carol are insufficient for Alice to obtain enough information to deduce $f(x,y,z)$. (This lower bound can be extended to all cases where $n>3$ by fixing the value of all but the first $n$ inputs of each party.)

First, consider the possibilities of which parties the two bits are sent among. Clearly there is no point in Alice sending the second bit. Also, if Alice sends the first bit to, say, Bob then there is no point in Carol sending the second bit to Alice (since the first bit sent is then useless to Alice). Therefore, if Alice sends the first bit to Bob then we can assume that Bob sends the second bit to Alice. Also, note that, by substituting Eq. (3) into Eq. (4),

$$f(x,y,z)=x_1\cdot y_1+x_2\cdot y_2+x_3\cdot y_3. \tag{10}$$

Thus, since only Alice and Bob are involved in the communication, this scenario reduces to the two-party inner product function, whose communication complexity is known to be three. Therefore there is no protocol in which Alice sends one of the two bits to Bob. Also, if Bob sends two bits to Alice then this can again be viewed as a two-bit two-party protocol computing Eq. (10), which is impossible. The above arguments also apply with Carol substituted for Bob.

The remaining possibilities are that Bob and Carol each send a single bit to Alice, or Bob sends a bit to Carol, who sends a bit to Alice (or vice versa). Both of these are subsumed by the scenario where Bob is allowed to broadcast one bit to both Alice and Carol, and then Carol sends one bit to Alice, who must output $f(x,y,z)$. This is the interesting case to examine.

The bit that Bob broadcasts is some function $\phi{:}\{0,1\}^3\rightarrow\{0,1\}$ of his input data $y$ alone. The function $\phi$ partitions $\{0,1\}^3$ into two classes $\phi^{-1}(0)$ and $\phi^{-1}(1)$. Call these two classes $S_0$ and $S_1$, and assume (without loss of generality) that $000\in S_0$. After Bob broadcasts his bit, what Alice and Carol each learn is whether $y\in S_0$ or $y\in S_1$. For a two-bit protocol to be correct, it must always be possible at this stage for Carol to send one bit to Alice that will enable Alice to completely determine the value of $f(x,y,z)$. We shall show that, whatever the partitioning $S_0,S_1$ is, there is an instance where Alice cannot determine the value of $f(x,y,z)$. There are 128 different possible partitionings, and each is one of the seven types that are examined below.

*Case 1* ($|S_0|\leqslant 2$): Recall our convention that $000\in S_0$. If $S_0$ has a second element then, by symmetry, no generality is lost if we assume that it is either 100, 110, or 111.

Thus, without loss of generality, $001{,}010{,}011\in S_1$. Now, should the bit that Bob broadcasts specify to Alice and Carol that $y\in S_1$, Carol must send one bit to Alice from which Alice can completely determine the value of $f(x,y,z)$.The bit that Carol sends induces a partition of the possible values of $z$ into two classes. If $x=001$ then, from Alice's perspective, after receiving Bob's bit but before receiving Carol's bit, the possible values of $(x,y,z)$ include (001,001,111), (001,010,100), (001,011,101), and the respective values of $f(x,y,z)$ on these points are 1,0,1. Therefore, for the protocol to be successful in this case, the partition that Carol's bit induces in $z$ must place 111 and 101 together in one class and 100 in the other class [otherwise Alice would not be able to determine $f(x,y,z)$ when $x=001$]. On the other hand, if $x=011$ then, from Alice's perspective, the possible values of $(x,y,z)$ include (011,001,101), (011,010,110), (011,011,111), and the respective values of $f(x,y,z)$ on these points are 1, 1, 0. Since we have established that Carol's bit does not distinguish between $z=111$ and $z=101$, Carol's bit is not sufficient information for Alice to determine $f(x,y,z)$ in this case.

*Case 2* ($|S_0|\geqslant 3$): For this case, we consider the subcases where either $S_0$ contains a string of weight 1 (i.e., that has exactly one 1) or does not.

*Case 2.1* ($|S_0|$ *contains a string of weight 1*): Without loss of generality, assume $001\in S_0$. By our convention, $000\in S_0$, and, after disregarding the obvious symmetries, there are four distinct possibilities for a third element of $S_0$: 010, 011, 110, 111 and these are considered separately.

*Case 2.1.1* ($000,\ 001,\ 010\in S_0$): The argument is similar to that in Case 1 using $S_0$ instead of $S_1$. Consider Alice's perspective. If $x=001$ then, the possible values for $(x,y,z)$ include (001,000,110), (001,001,111), (001,010,100) for which the respective values of $f(x,y,z)$ are 0, 1, 0; whereas, if $x=011$ then the possible values for $(x,y,z)$ include (011,000,100), (011,001,101), (011,010,110) for which the respective values of $f(x,y,z)$ are 0, 1, 1. No binary partitioning of $z$ will work for both possibilities.

*Case 2.1.2* ($000,\ 001,\ 011\in S_0$): Consider Alice's perspective. If $x=001$ then, the possible values for $(x,y,z)$ include (001,000,110), (001,001,111), (001,011,101) for which the respective values of $f(x,y,z)$ are 0, 1, 1; whereas, if $x=011$ then the possible values for $(x,y,z)$ include (011,000,100), (011,001,101), (011,011,111) for which the respective values of $f(x,y,z)$ are 0, 1, 0. No binary partitioning of $z$ will work for both possibilities.

*Case 2.1.3* ($000,\ 001,\ 110\in S_0$): Consider Alice's perspective. If $x=010$ then, the possible values for $(x,y,z)$ include (010,000,101), (010,001,100), (010,110,011) for which the respective values of $f(x,y,z)$ are 0, 0, 1; whereas, if $x=011$ then the possible values for $(x,y,z)$ include (011,000,100), (011,001,101), (011,110,010) for which the respective values of $f(x,y,z)$ are 0, 1, 1. No binary partitioning of $z$ will work for both possibilities.

*Case 2.1.4* ($000,\ 001,\ 111\in S_0$): Consider Alice's perspective. If $x=010$ then, the possible values for $(x,y,z)$ include (010,000,101), (010,001,100), (010,111,010) for which the respective values of $f(x,y,z)$ are 0, 0, 1; whereas, if $x=011$ then the possible values for $(x,y,z)$ include (011,000,100), (011,001,101), (011,111,011) for which the respective values of $f(x,y,z)$ are 0, 1, 0. No binary partitioning of $z$ will work for both possibilities.

*Case 2.2* ($|S_0|$ *contains no string of weight 1)*: We consider the following three subcases.

*Case 2.2.1* ($111\notin S_0$): In this case, $011{,}010{,}100{,}111\in S_1$. Suppose that Bob's bit specifies that $y\in S_1$. Consider Alice's perspective. If $x=001$ then, the possible values for $(x,y,z)$ include (001,001,111), (001,010,100), (001,100, 010), (001,111,001) for which the respective values of $f(x,y,z)$ are 1, 0, 0, 1; whereas, if $x=010$ then the possible values for $(x,y,z)$ include (010,001,100), (010,010,111), (010,100,001), (010,111,010) for which respective values of $f(x,y,z)$ are 0, 1, 0, 1. No binary partitioning of $z$ will work for both possibilities.

*Case 2.2.2* ($111\in S_0$): In this case, $S_0$ must contain an element of weight 2. Without loss of generality, $011\in S_0$. Therefore, $000{,}011{,}111\in S_0$. Consider Alice's perspective. If $x=010$ then, the possible values for $(x,y,z)$ include (010,000,101), (010,011,110), (010,111,010) for which the respective values of $f(x,y,z)$ are 0, 1, 1; whereas, if $x=110$ then the possible values for $(x,y,z)$ include (110,000,001), (110,011,010), (110,111,110) for which the respective values of $f(x,y,z)$ are 0, 1, 0. No binary partitioning of $z$ will work for both possibilities.

This concludes the proof that there is no classical protocol for computing $f(x,y,z)$ in which only two bits are communicated among Alice, Bob, and Carol.

## A THREE-BIT CLASSICAL PROTOCOL

Although one might suspect that, without the use of entangled particles, $n$ bits of communication are necessary for Alice to determine $f(x,y,z)$ in general, it turns out that three bits always suffice.

The idea behind the method is to count the total number of 0's among all the $3n$ inputs of Alice, Bob, and Carol. Note that, for each $i\in\{1,\ldots,n\}$, if $x_i\cdot y_i\cdot y_i=1$ then there are zero 0's among $x_i,y_i,z_i$, and if $x_i\cdot y_i\cdot y_i=0$ then there are two 0's among $x_i,y_i,z_i$.Let the number of 0's among $x_1,\ldots,x_n$ be $r_A$, the number of 0's among $y_1,\ldots,y_n$ be $r_B$, and the number of 0's among $z_1,\ldots,z_n$ be $r_C$. Let $k$ be the total number of terms among $x_1\cdot y_1\cdot z_1,\ldots,x_n\cdot y_n\cdot z_n$ that have value 0. Then, from the above, $r_A+r_B+r_C=2k$.

Therefore, it suffices for Bob to send $r_B$ to Alice and Carol to send $r_C$ to Alice in order for Alice to compute $k$. From $k$, Alice can easily compute $f(x,y,z)=(n-k)\mathrm{mod}2$. This involves $2\log_2 n$ bits of communication. Fortnow [8] has shown that the communication can be reduced to three bits as follows. Since Alice only needs the parity of $k$, she only needs the values of $r_A$, $r_B$, $r_C$ in modulo 4 arithmetic. Therefore, it suffices for Bob and Carol to each send two bits to Alice. This yields a four-bit protocol. To obtain a three-bit protocol, note that $r_A+r_B+r_C$ is guaranteed to be an even number. This means that either Bob or Carol can send just the high order bit of his or her two-bit number.

## ACKNOWLEDGMENTS

We would like to thank Lance Fortnow for allowing us to incorporate his improvement to the $2\log_2$n-bit classical protocol. R.C. is grateful to Lev Vaidman for introducing him to the three-particle deterministic examples of Bell nonlocality, Richard Jozsa for providing references to this work, and David DiVincenzo for general discussions about the power of entanglement. R.C. is supported in part by Canada's NSERC. H.B. is supported by NWO by SION Project 612-34-002, EU through Neuro COLT ESPRIT Working Group No. 8556, and HC&M Grant CCR 92-09833.

[1] A. Einstein, B. Podolsky, and N. Rosen, Phys. Rev. **47**, 777 (1935).

[2] J. S. Bell, Physics (Long Island City, NY) **1**, 195 (1964).

[3] J. F. Clauser, M. A. Horne, A. Shimony, and R. A. Holt, Phys. Rev. Lett. **23**, 1881 (1969).

[4] D. M. Greenberger, M. Horne, and A. Zeilinger, in *Bell's Theorem, Quantum Theory, and Conceptions of the Universe*, edited by M. Kafatos (Kluwer Academic, Dordrecht, 1989), p. 69.

[5] N. D. Mermin, Phys. Today **43** (6), 9 (1990); Am. J. Phys. **58**, 731 (1990).

[6] E. Kushilevitz and N. Nisan, *Communication Complexity* (Cambridge University Press, Cambridge, 1996); the proof that the communication complexity of the inner product function is $n$ is in Chapter 1: Exercise 1.29 and the remark immediately preceding it on p. 14 imply a lower bound of $n+1$ in the model where both parties must acquire the answer; this, in combination with a remark on p. 5, implies a lower bound of $n$ in the model where only Alice must acquire the answer.

[7] L. K. Grover, e-print Quantum Telecomputation, Report No. quant-ph/9704012, 1997 (unpublished).

[8] L. Fortnow (private communication).