Quantum Information Processing (2022) 21:151
https://doi.org/10.1007/s11128-022-03492-y

Quantum Byzantine Agreement for Any Number of Dishonest Parties

Vicent Cholvi[1]

Received: 22 December 2021 / Accepted: 11 March 2022 / Published online: 7 April 2022

Abstract

Reaching agreement in the presence of arbitrary faults is a fundamental problem in distributed computation, which has been shown to be unsolvable if one-third of the processes can fail, unless signed messages are used. In this paper, we propose a solution to a variation of the original BA problem, called Detectable Byzantine Agreement (DBA), that does not need to use signed messages. The proposed algorithm uses what we call *Q-correlated lists*, which are generated by a *quantum source device*. Once each process has one of these lists, they use them to reach the agreement in a classical manner. Although, in general, the agreement is reached by using $m + 1$ rounds (where m is the number of processes that can fail), if less than one-third of the processes fail it only needs one round to reach the agreement.

Keywords Byzantine agreement · Quantum information · Quantum distribution algorithms · Quantum communication · Q-Correlated lists

1 Introduction

Reaching agreement in the presence of arbitrary faults is a fundamental problem in distributed computation, which has been extensively studied in the past. This problem, also called as Byzantine agreement (BA), consists of several Byzantine generals who are commanding their army divisions to besiege an enemy city. They must decide upon a common plan of action, but they can communicate with one another only by pairwise error-free classical channels. One of the generals, the *commanding general*, must decide on a plan of action and communicate it to the other generals. However, some of the generals, including the *commander* can be dishonest and try to prevent

✉ Vicent Cholvi
vcholvi@uji.es

[1] Departament de Llenguatges i Sistemes Informàtics, Universitat Jaume I, Campus Rius Sec s/n, Castelló 12071, Spain

the honest generals from reaching agreement of the plan of action. Thus, the solution to the problem must satisfy:

$IC1$: All honest parties obey the same order.
$IC2$: If the commanding general is honest, then every honest party obeys the order he sends.

In [1], it was shown that this problem is unsolvable if one-third of the generals are dishonest. In [2], the authors provided a solution that works for any number of dishonest generals. However, this algorithm and all the subsequent ones that works for any number of dishonest generals, require an authentication structure based on signed messages (e.g., [3]).

On another hand, in [4] the authors proposed a variation of the original BA problem, called Detectable Byzantine Agreement (DBA), which relaxes the above-mentioned IC1 and IC2 conditions so that all honest parties either perform the same action or all abort. The advantage of using DBA instead of BA is to avoid the use of signed messages and, as it has been argued in [4], using DBA is enough for applications where robust tolerance to errors is not necessary and detection suffices.

The authors in [4–6] presented quantum solutions to the DBA problem but for only three parties (the *commander* and two generals). In [7], a quantum solution has been proposed that considers any number of parties, but it assumes that less than one-third of the parties will be dishonest. Another quantum solution that considers any number of parties has been presented in [8], but also assumes that less than one-third of the parties will be dishonest. As far as we know, there have been only two proposals to solve the DBA problem for any number of dishonest parties [9, 10], but their agreement solutions are not fully correct (see Sect. 5).

Our work
In this paper, we propose a solution for the DBA problem, without using signed messages, for any number of dishonest generals, which we call *parties*. For this task, we use *Q-correlated lists*. Such lists are distributed to the parties by using a number of entangled quantum particles that are generated by a *quantum source device*. Once each party has one of these lists, they use them to reach the agreement in a classical manner. At this point, our proposed solution has two interesting features:

1. On one hand, any forgery of the state of the above-mentioned particles (and, therefore, in the Q-correlated lists) can be detected.
2. On the other hand, the option of abort is considered only in the distribution of the lists. Thus, in the agreement phase, our solution still enables full BA.

The rest of the paper is structured as follows. In Sect. 2, we define Q-correlated lists, In Sect. 3, we show how the above-mentioned Q-correlated lists can be distributed, so that any forgery of their states can be detected. In Sect. 4, we introduce an algorithm that, by using these lists, solves the BA problem in a classical manner without using any quantum resources. We end, in Sect. 6, with some open issues.

2 Sets of Q-correlated lists

In this section, we introduce a data structure, which we call *Q-correlated list*, that is the core of the BA algorithm presented in Sect. 4. In Sect. 3, we will show how, by using a number of entangled quantum particles, it is possible to provide each party (including the *commander*) with one of the above-mentioned list.

Given a list L, we denote as L^k the element at position k in the list L.

Definition 1 Let $\mathcal{S} = \{L_1, ..., L_n\}$ be a set of n lists, each formed by elements in $W = \{0, 1, \ldots, w\}$, with $w \geq n$. We say that $\mathcal{S}$ is *Q-correlated* (where Q is a set of positions in the lists) if the following three conditions hold:

1. All the lists have the same length.
2. All the elements are random values in W.
3. For each two different L_i and L_j in $\mathcal{S} : L_i^k \neq L_j^k$, provided $k \in Q$.

The positions in Q are called *correlated* positions. Observe that the elements at position k in these lists (i.e., $L_1^k L_2^k \ldots L_n^k$) are either (1) different random numbers in W if k is a correlated position, or (2) random numbers in W if k is not a correlated position (although these numbers may be different). Note that, since the number of elements in W is greater than the number of lists in $\mathcal{S}$, from a subset of lists is not possible to infer, with complete certainty, what the others will be, even if it is known which positions are correlated.

Example Let $\mathcal{S} = \{\{1, 2, 0, 0, 3, 2, 3\}, \{2, 1, 3, 0, 0, 0, 2\}, \{0, 3, 1, 3, 1, 1, 0\}, \{3, 0, 2, 2, 2, 3, 1\}\}$, with $W = \{0, 1, 2, 3\}$. $\mathcal{S}$ is Q-correlated with $Q = \{1, 2, 3, 5, 6, 7\}$, since all the lists have the same length and, at the same correlated positions, the elements take different values. On the contrary, $\mathcal{S}$ is not Q-correlated with $Q = \{3, 4, 5\}$, since the fourth element is the same in the first and second lists.

Definition 2 Let $v \in W = \{0, 1, \ldots, w\}$ and let $\mathcal{L}$ be a set of lists each formed by elements in W. We say that the pair $(v, \mathcal{L})$ is *consistent* provided the following three conditions hold:

1. All the lists in $\mathcal{L}$ have the same length.
2. All the elements in the lists in $\mathcal{L}$ are random values in $W - \{v\}$.
3. For each two lists $\mathcal{L}_i$ and $\mathcal{L}_j$ in $\mathcal{L} : \mathcal{L}_i^k \neq \mathcal{L}_j^k$, for all k.

Next, we will state two properties of the Q-correlated sets of lists that will be key in the operation of the proposed agreement algorithm. Given a set of positions R, we denote as L^R the list formed by the elements L^k such that $k \in R$, maintaining these elements in the same relative order as in L. Note that L^R denotes a list of elements, whereas L^k denotes an element.

Property 1 *Let $\mathcal{S}$ be a Q-correlated set of lists, each formed by elements in W. Let $v \in W$ and L_i an arbitrary list in $\mathcal{S}$. Let $R \subseteq Q$ such that $L_i^k = v$ for all $k \in P$, and $\mathcal{L}$ a set of lists of the form L_j^R, where $j \neq i$. The pair $(v, \mathcal{L})$ is consistent,*

Proof Clearly, all the lists in $\mathcal{L}$ have the same length. Since $\mathcal{S}$ is Q-correlated then the elements at the same positions in the lists in $\mathcal{L}$ are different. Furthermore, these values will be different from v (since v appears in L_i^R in all positions). Therefore, the obtained pair will be consistent. □

Example By using the previous set $\mathcal{S}$ with $Q = \{1, 2, 3, 5, 6, 7\}$, if we know the values of the list L_1 then, for $v = 2$, we can choose $R = \{2, 6\}$ and we guarantee that any pair $(2, \mathcal{L})$ (with $\mathcal{L}$ formed by L_j^R lists, where $j \neq 1$) is consistent.

Property 2 *Let $\mathcal{S}$ be a Q-correlated set of lists, each formed by elements in W. Assume that we don't know the values of some arbitrary list $L_i \in \mathcal{S}$ and which positions are correlated. Then, it is not possible to choose a set of lists $\mathcal{L}$ (not necessarily in $\mathcal{S}$), each formed by elements in W, and a set R of positions in these lists, such that the pair $(v, \mathcal{L}')$ where $\mathcal{L}' \equiv \{\mathcal{L}, L_i^R\}$ is guaranteed to be consistent.*

Proof Since the number of elements in W is greater than the number of lists in $\mathcal{S}$, we cannot identify with complete certainty which are all the correlated positions, even if we know the values of all the lists in $\mathcal{S}$, except L_i.

Then, assume that we choose R such that it contains a non-correlated position k. Since that position is non-correlated, we are not guaranteed that the value at position k in L_i^R won't be v, or any of the values at position k in the lists in $\mathcal{L}$, which will make the pair $(v, \mathcal{L}')$ inconsistent. In other words, we cannot fully guarantee that the pair $(v, \mathcal{L}')$ will be consistent. □

3 Distributing the *Q*-correlated lists

For the distribution of the Q-correlated lists among the parties, we assume that there is an honest independent *quantum source device* (QSD) that will communicate with the parties through pairwise error-free quantum channels. A pairwise quantum channel is said to be *error-free* provided it guarantees that there will be no change in the state of any sent particle due to the own channel, although there is no guarantee that such state could be tampered by third parties. That QSD will prepare and distribute a number of particles so that each party, by measuring them, will obtain one list Q-correlated with the other parties' lists.

Let $W = \{0, 1, \ldots, w\}$, with $w \geq n$ (where n is the number of parties). The particles that will be distributed are of three types:

1. Particles in the following uniform random states: $|\Psi_0\rangle = \frac{1}{\sqrt{w+1}} \sum_{j=0}^{w} |j\rangle$. Clearly, the measured states of each particle will obtain a random uniform value in W.
2. Particles in the following quantum entangled states: $|\Psi_1\rangle = \frac{1}{\sqrt{w+1}} \sum_{j=0}^{w} |j \otimes j\rangle$. Now, the measured states of each one single-particle will obtain the same value in W.
3. Particles in the following quantum entangled states:

1. Let $W = \{0, 1, \cdots, w\}$, so that $w \geq n$ (where n is the number of parties).
2. For $t = 1$ to L, where L denotes the length of the lists, the QSD decides whether position t in the lists will be correlated or not (that decision is taken at random):
   (a) If position t is chosen to be correlated then the QSD prepares q particles in the entangled state $\left|\Psi^{0}_{i_1,i_2,\cdots,i_w}\right\rangle_{w+1}$ by taking parameters with different values. Then, the QSD sends one particle to each party except the *commander*, to whom it sends two particles.
   (b) If position t is chosen to be non-correlated:
      (i) The QSD prepares and sends one particle in the state $|\Psi_0\rangle$ to each party, except the *commander*.
      (ii) The QSD prepares two particles in the entangled state $|\Psi_1\rangle$ and sends them to the *commander*.
3. Once all the particles have been received by the parties:
   (a) Each party (except the *commander*) will measure their state and will generate a list with the obtained values.
   (b) The *commander* will measure their state and use the first particles of each received pair to generate its list. In addition, it will the use the second particles to detect whether a positions is correlated or not: namely, a position is correlated when the values of each received pair of particles is different.

Fig. 1 The algorithm to distribute the Q-correlated lists

$$\left|\Psi^{s}_{i_1,i_2,\ldots,i_{q-1}}\right\rangle_q = \frac{1}{\sqrt{d}}\sum_{j=0}^{d-1} e^{\frac{2\pi ijs}{d}}|j\rangle \otimes |j+i_1 \mod d\rangle \otimes \ldots \otimes |j+i_{q-1} \mod d\rangle,$$

where $q, i_1, \ldots, i_q \in \{0, 1, \ldots, d-1\}$. If we take $s = 0$ then we have:

$$\left|\Psi^{0}_{i_1,i_2,\ldots,i_{q-1}}\right\rangle_q = \frac{1}{\sqrt{d}}\sum_{j=0}^{d-1} |j\rangle \otimes |j+i_1 \mod d\rangle \otimes \ldots \otimes |j+i_{q-1} \mod d\rangle.$$

Let us also we take $q = d = w + 1$ and let us perform the measurements of the single-particle states in the base $MB = \{|0\rangle, |1\rangle, \ldots, |w\rangle\}$, denoting the measured state $|0\rangle$ as 0, $|1\rangle$ as 1, ..., $|w\rangle$ as w. As it has been shown in [11], if the parameters $i_1, \ldots, i_w$ in $\left|\Psi^{0}_{i_1,i_2,\ldots,i_w}\right\rangle_{w+1}$ are different then each one of the $w + 1$ single-particle measured states will obtain a different value in W.

Figure 1 shows the basic distribution process. Particles of types 1 and 2 will be used to provide uncorrelated values (so, we will call the uncorrelated particles), whereas particles of type 3 will be used to provide correlated ones (and we will call them correlated particles). This is because particles of type 3 are the only ones that guarantee

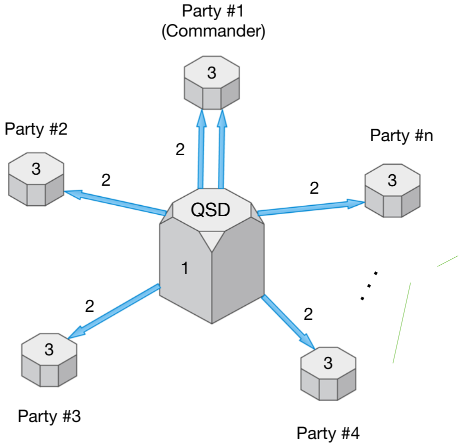

Fig. 2 Scheme of the quantum protocol used to distribute the Q-correlated list of length L. *1* The QSD prepares $n+1$ particles, either correlated or non-correlated. *2* The QSD sends one of the previously prepared particles to each party, except for the *commander* to whom it sends two particles. Steps 1 and 2 are repeated consecutively for L times. *3* Once all the particles have been transmitted, each party measures the states of the received particles and generates its list. As a result, correlated particles will generate correlated values, whereas non-correlated particles will generate non-correlated values. In addition, the *commander* uses the second particles received to detect which positions are correlated (namely, when the values are different from the first particles received)

that, when measured, their values will be different. While the values provided by particles of type 2 will be always the same, the values provided by particles of type 1 may or may not be different; however, if we use a large enough number of particles, we will guarantee with high probability that there will be some case where the values measured by two parties will be equal.

To further clarify how the above-mentioned quantum protocol works, in Fig. 2 we show how it interacts with both the QSD and the parties.

Checking the presence of eavesdroppers

Although for the distribution of the Q-correlated lists it has been assumed that an honest QSD generates the particles, which are send to the parties through pairwise error-free quantum channels, if anyone obtains information about what the QSD transmits (e.g., the correlated positions or the state of the transmitted particles), such information could be used to generate consistent data and, therefore, to break the subsequent agreement process.

In our work, we will integrate the eavesdroppers detection into our basic distribution protocol by taking an approach similar to the one used in [8], which is based on making use of a *quantum private comparison* (QPC) protocol [12–14] to prevent particles from being tampered.

Roughly speaking, it consists of making the QSD to generate a number of *decoy* particles, and insert them, at random, into the sent sequences of particles. The key feature is that these decoy particles will be generated by using two unbiased orthog-

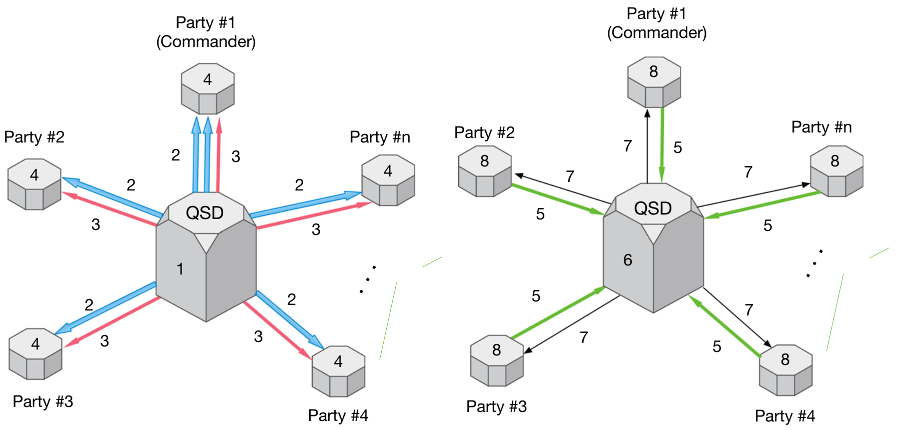

Fig. 3 Scheme of the quantum protocol used to distribute the Q-correlated list of length L with eavesdroppers detection. *1* The QSD prepares $n + 1$ particles, either correlated, non-correlated or decoy. *2* The QSD sends one of the previously prepared particles to each party, except for the *commander* to whom it sends two particles. Steps 1 and 2 are repeated consecutively for L times. *3* Once all the particles have been transmitted, the QSD sends the positions and the bases of the decoy particles. *4* The parties measure the decoy particles using the corresponding bases. *5.* The parties return the measured results to the QSD. *6.* The QSD checks whether eavesdroppers exist in the quantum channels or not. *7.* The QSD communicates whether the distribution protocol is aborted or not. *8.* If the distribution protocol is not aborted, each party measures the states of the received non-decoy particles and generates its list. In addition, the *commander* uses the second particles received to detect which positions are correlated

onal bases: namely, the previously defined MB and $MF = \{F|0\rangle, F|1\rangle, \ldots, F|w\rangle\}$, where F is the discrete Fourier transform. By taking that into account, the QSD will only announce (to the parties) the position and bases of the decoy particles after all the particles have been transmitted. Then, the parties will measure these particles and will return the results to the QSD, which will verify these results and will check whether eavesdroppers exist in the quantum channels or not. The details of the verification process can be found in [14]. If eavesdroppers are found, the protocol aborts. Otherwise, the parties generate their Q-correlated lists and execute the agreement algorithm described in the next section.

To further clarify how the above-mentioned quantum protocol works, in Fig. 3 we show how the eavesdroppers detection is integrated into our basic distribution protocol. The figure on the left shows how it works until the parties measure the decoy particles, and the figure on the right shows from then until the parties generate their lists.

4 The *QBA*(*m*) algorithm

By using the algorithm introduced in the previous section, we can guarantee that each party will have one list of a Q-correlated set. Now, in this section, we introduce an algorithm that, by using these lists, solves the BA problem in a classical manner without using any quantum resources.

The code of the above-mentioned algorithm, which we called $QBA(m)$, is shown in Fig. 4. It assumes that the parties can communicate with one another by pairwise *safe*

```
1. Use the algorithm in Figure 1 to distribute among the parties a set of $Q$-correlated lists of *sufficiently long* length. As a result, we have that:
   (a) Each party has one list in a $Q$-correlated set of lists.
   (b) The *commander* is the only party that knows which are the correlated positions.
2. Let $v \in W$ be the order to be transmitted by the *commander* $c$ and let $\mathcal{L} = \{\}$. Then, he sends $(P, (v, \mathcal{L}))$ to each party $i$ through pairwise error-free classical channels, where $P$ is a list of correlated positions in $L_c$ in which $v$ appears (but not necessarily all the positions).
3. For each party $i$ (except for the *commander*):
   (a) If it receives $(P, (v, \mathcal{L}))$ from the *commander*:
       (i) Add $L_i^P$ to $\mathcal{L}$.
       (ii) If $(v, \mathcal{L})$ is consistent then:
            (A) $V_i = v$
            (B) Send $(P, (v, \mathcal{L}))$ to all the parties.
   (b) For $m + 1$ rounds (starting at round 1), in each round perform: if at round $r$ it receives $(P, (v, \mathcal{L}))$:
       (i) Add $L_i^P$ to $\mathcal{L}$.
       (ii) If $(v, \mathcal{L})$ is consistent, $v \notin V_i$ and the number of lists in $\mathcal{L}$ is $r + 1$:
            (A) Add $v$ to $V_i$.
            (B) If $r \leq m$ then send $(P, (v, \mathcal{L}))$ to all the parties.
   (c) $V_i$ will be the same for all the honest parties, so they can decide the same.
```

Fig. 4 The $QBA(m)$ algorithm for m dishonest parties

classical channels. Namely, we say that a classical channel is safe provided (i) every message that is sent is delivered correctly, (ii) the receiver of a message knows who sent it and (iii) the absence of a message can be detected. However, since parties (including the *commander*) can be dishonest, they can send consistent or inconsistent data (see Definition 2). This includes the case where one dishonest party sends consistent data to some parties and inconsistent data (or no data) to the rest.

As it can be seen in the Step 1 of the algorithm, we require that the distributed lists are of *sufficiently long* length. This requirement is introduced in order to avoid any casually created consistent pair, which can be guaranteed with high probability as we increase the length of the lists.

Theorem 1 *The protocol $QBA(m)$ solves with high probability the Byzantine Agreement problem for m dishonest parties.*

Proof *Prove IC2:* assume the *commander* is honest. So, every party will receive the same data from the *commander*. Since no dishonest party can forge that data so that it also looks consistent (by Property 2 and taking into account that the *commander* is the only one party that knows which positions are correlated), by Property 1, the set V_i (for each i) will always contain the same and unique value sent by the *commander*. Therefore, all honest parties [at step 3(c)] will decide the value sent by the *commander*.

Prove IC1: assume the *commander* is dishonest. Two honest parties i and j decide the same provided V_i and V_j are the same when they take the decision [i.e., at step 3(c)]. Therefore, we only need to prove that if i adds v to V_i then j also adds v to V_j. That is, we have to show that j will also receive a consistent tuple with the value v.

1. If i receives that value at step 3(a) then it sends it to j in step 3(a)iiB, who will add it to V_j [at step 3(b)iiA].
2. If i adds v to V_i at step 3(b)i then that's because it received at that round consistent data for that value. Now, we have two possibilities:
    - Party i receives the data before round $m + 1$: in this case, i will send that value to j [at step 3(b)iiB], who will add it to V_j [at step 3(b)iiA].
    - Party i receives the data at round $m + 1$: in this case, party i won't send any data and, therefore, party j won't receive data with that value. Since there is, at most, m dishonest parties, to consider consistent data at round $m + 1$, such a data must contain $m + 1$ lists. However, all lists in $\mathcal{L}$ different that L_x^P will make that data inconsistent. Indeed, let's assume that we add a list L' different from L_x^P. Let v' be a value that appears at position k in list L'. We know that, at that position, there will be different values in the other parties' list (assuming that we know that it is a correlated position; otherwise is even simpler). However, we don't know the concrete values, at that position, in all the other parties' lists (note that $w \geq n$); so, it could happen that v' appears in another list at the same position, which will certainly happen if P is long enough. Therefore, the addition of L' to $\mathcal{L}$ will make the pair inconsistent. Consequently, one of the lists in $\mathcal{L}$ (i.e., L_x^P) must be from an honest party, who will have sent consistent data with the value v to all the parties before round $m + 1$. Thus, v will be already included both in V_i and V_j.

This completes the proof. □

We would like to note that, for the sake of clarity, we have presented our BA algorithm as simple as possible. However, it can be optimized in some cases. For instance:

1. Our algorithm requires $m + 1$ rounds to finish, but it can be easily adapted to the case where $m < n/3$, so that the decision is made by using only one round (the approach is similar to that in [8]).
2. If the absence of messages can be detected, then it is possible to advance the decision making immediately after detecting that no message has been transmitted at a given round.

5 Previous solutions of the DBA problem for any number of dishonest parties

As it has been mentioned in the introduction, there have been two previous proposals to solve the DBA problem for any number of dishonest parties. Here we show, by means of two counter-examples, that their agreement solutions are not fully correct.

- Takavoli et al. [9]: this algorithm is intended to solve binary DBA. In the algorithm in Table 1, assume P_1 is faulty and sends consistent pairs to all the processes, so that all messages are the same, except one. Now assume that the process that receives the different message (which is also faulty) conveys its received pair to some processes, and $\perp$ to the rest: the processes that receive the pair will decide to abort (since they detect, by (iib), that P_1 is faulty), but those who receive $\perp$ will decide the value sent by P_1 [they apply (iid)]. That is, non-faulty processes will decide different things. Furthermore, the quantum protocol used for distributing the correlated lists has not been shown to be always correct. For instance, it could happen that a dishonest process reveals a fake encoding base (e.g., choosing it at random) so that, by chance, the sum of the basis choices modulo m equals zero, while the sum of the right basis choices modulo m is different from zero. In that case, the run would be treated as a valid distribution of the numbers at the same position in the private lists. That is enough to break the subsequent Byzantine agreement algorithm.
- Sun et al. [10]: this algorithm is intended to solve multivalued DBA. At stage 2, assume that P_1 is faulty and sends consistent pairs to all the processes, so that all messages are the same, except one. Now, assume that the process that receives the different value (which is also faulty) conveys its received pair to some processes, and $\perp$ to the rest: the processes that receive the consistent pair will decide $\perp$ [they will apply 3(a)], but those who receive $\perp$ will decide the value send by P_1 [(they will apply 3(c)]. That is, non-faulty processes will decide different things.

6 Open issues

1. Whereas in this paper we assumed that the QSD is an independent device, perhaps the parties themselves could be used to generate and send the particles. This technique has already been used by Gaertner et al. [5] in the case of three parties.
2. Based on Hardy's correlations [15] and entanglement swapping, the authors in [16] have presented a protocol for the original BA problem with three parties. So, maybe that could also be used to avoid the possibility of abortion, during the distribution process, when considering several parties.

Acknowledgements This work has been partially funded by the Spanish Ministry of Science and Innovation grant PID2019-109805RB-I00 (ECID) cofunded by FEDER.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

Availability of data and materials Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

References

1. Pease, M., Shostak, R., Lamport, L.: Reaching agreement in the presence of faults. J. ACM **27**(2), 228–234 (1980). https://doi.org/10.1145/322186.322188
2. Lamport, L., Shostak, R., Pease, M.: The byzantine generals problem. ACM Trans. Progr. Lang. Syst. **4**(3), 382–401 (1982). https://doi.org/10.1145/357172.357176
3. Dolev, D., Strong, H.R.: Authenticated algorithms for byzantine agreement. SIAM J. Comput. **12**(4), 656–666 (1983)
4. Fitzi, M., Gisin, N., Maurer, U.: Quantum solution to the byzantine agreement problem. Phys. Rev. Lett. **87**, 217–901 (2001). https://doi.org/10.1103/PhysRevLett.87.217901
5. Gaertner, S., Kurtsiefer, C., Bourennane, M., Weinfurter, H.: Experimental demonstration of four-party quantum secret sharing. Phys. Rev. Lett. **98**, 020–503 (2007). https://doi.org/10.1103/PhysRevLett.98.020503
6. Gaertner, S., Bourennane, M., Kurtsiefer, C., Cabello, A., Weinfurter, H.: Experimental demonstration of a quantum protocol for byzantine agreement and liar detection. Phys. Rev. Lett. **100**, 070–504 (2008). https://doi.org/10.1103/PhysRevLett.100.070504
7. Ben-Or, M., Hassidim, A.: Fast quantum byzantine agreement. In: Proceedings of the 37th Annual ACM Symposium on Theory of Computing, STOC'05, pp. 481–485. Association for Computing Machinery, New York (2005). https://doi.org/10.1145/1060590.1060662
8. Luo, Qb., Feng, Ky., Zheng, Mh.: Quantum multi-valued byzantine agreement based on d-dimensional entangled states. Int. J. Theor. Phys. **58**(12), 4025–4032 (2019). https://doi.org/10.1007/s10773-019-04269-3
9. Tavakoli, A., Cabello, A., Zukowski, M., Bourennane, M.: Quantum clock synchronization with a single qudit. Sci. Rep. **5**, 7982 (2015). https://doi.org/10.1038/srep07982
10. Sun, X., Kulicki, P., Sopek, M.: Multi-party quantum byzantine agreement without entanglement. Entropy **22**(10), 1152 (2020). https://doi.org/10.3390/e22101152
11. Liu, X.S., Long, G.L., Tong, D.M., Li, F.: General scheme for superdense coding between multiparties. Phys. Rev. A **65**, 022–304 (2002). https://doi.org/10.1103/PhysRevA.65.022304
12. Chen, X., Xu, G., Niu, X., Wen, Q., Yang, Y.: An efficient protocol for the private comparison of equal information based on the triplet entangled state and single-particle measurement. Opt. Commun. **283**, 1561–1565 (2010)
13. Yang, Y.G., Wen, Q.Y.: An efficient two-party quantum private comparison protocol with decoy photons and two-photon entanglement. J. Phys. A—Math. Theor. (2009). https://doi.org/10.1088/1751-8113/42/5/055305
14. Chang, Y.J., Tsai, C.W., Hwang, T.: Multi-user private comparison protocol using GHZ class states. Quantum Inf. Process. **12**(2), 1077–1088 (2013). https://doi.org/10.1007/s11128-012-0454-z
15. Hardy, L.: Quantum mechanics, local realistic theories, and Lorentz-invariant realistic theories. Phys. Rev. Lett. **68**, 2981–2984 (1992). https://doi.org/10.1103/PhysRevLett.68.2981
16. Rahaman, R., Wieśniak, M., Żukowski, M.: Quantum byzantine agreement via hardy correlations and entanglement swapping. Phys. Rev. A **92**, 042–302 (2015). https://doi.org/10.1103/PhysRevA.92.042302

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