An extended abstract of this paper appeared in the proceedings of the *24th Symposium on Principles of Distributed Computing (PODC 2005)* July 17-20, 2005, Las Vegas, Nevada, USA, ACM Press. This is full version.

Simultaneous Broadcast Revisited

Alejandro Hevia* Daniele Micciancio†

April 2006

Abstract

Simultaneous Broadcast protocols allow different parties to broadcast values in parallel while guaranteeing mutual independence of the broadcast values. In this work, we study various definitions of independence proposed in the literature by Chor, Goldwasser, Micali and Awerbuch (FOCS 1985), Chor and Rabin (PODC 1987) and Gennaro (IEEE Trans. on Parallel and Distributed Systems, 2000), and prove implications and separations among them.

In summary, we show that each definition (generalized to allow arbitrary input distributions) is characterized by a class of "achievable" input distributions such that there is a single protocol that simultaneously meets the definition for all distributions in the class, while for any distribution outside the class no protocol can possibly achieve the definition. When comparing sets of achievable distributions, the definition of Gennaro is the most stringent (followed by the Chor and Rabin one, and Chor, Goldwasser, Micali and Awerbuch as the most relaxed) in the sense that it is achievable for the smallest class of distributions. This demonstrates that the definitions of Gennaro, and Chor and Rabin are of limited applicability.

Then, we compare the definitions when restricted to achievable distributions. This time the results of our comparison rank the definitions in the opposite order, with the definition of Chor, Goldwasser, Micali and Awerbuch as the strongest one (followed by Chor and Rabin, and then Gennaro) in the sense that security according to the stronger definitions implies security according to the weaker ones. We also give examples showing that the implications are strict, i.e., there are input distributions such that a protocol can meet the weaker definition, but fail to satisfy the stronger. The separation between the definitions of Gennaro and Chor and Rabin is particularly strong, as we show that there is a single protocol that is simultaneously secure according to Gennaro under any achievable input distribution, but does not satisfy the definition of Chor and Rabin for any non-trivial distribution. In particular, the separation holds for the special case of the uniform input distribution originally considered by the authors in their papers.

Keywords: Independence, Parallel Broadcast, Secure Function Evaluation.

*Dept. of Computer Science & Engineering, University of California, San Diego, 9500 Gilman Drive, La Jolla, California 92093, USA. E-mail: ahevia@cs.ucsd.edu, URL: http://www-cse.ucsd.edu/users/ahevia. Supported in part by grant of second author. On leave from the Dept. of Computer Science, University of Chile, Chile.

†Dept. of Computer Science & Engineering, University of California, San Diego, 9500 Gilman Drive, La Jolla, California 92093, USA. E-mail: daniele@cs.ucsd.edu, URL: http://www-cse.ucsd.edu/users/daniele. Supported also by a Sloan research fellowship.

Contents

1 Introduction

Broadcast channels allow one or more senders to efficiently transmit messages to be received by all parties connected to a (physical or virtual) communication network. Broadcast is a fundamental communication primitive, both in the design of network communication protocols, and in the area of secure multiparty computation. The main security property characterizing broadcast communication is consistency: the messages received by all players as a result of a broadcast transmission operation are guaranteed to be the same. The problem of achieving consistency when implementing broadcast on top of a point to point network (commonly known as the Byzantine agreement problem) is central not only in cryptography, but also to the area of fault-tolerant distributed computation, and it has received enormous attention (e.g., [LSP82, PSL80, FM85, CR93, CKPS01]).
In secure multiparty computation, it is often desirable that the broadcast channel satisfies some additional properties, besides consistency. In applications where multiple senders can broadcast messages at the same time (e.g., when running in parallel many copies of a broadcast protocol with different senders), it is often important to enforce the *simultaneous* transmission of the messages, so that no sender can decide its broadcast message based on the values broadcast by the other players. This independence property plays a fundamental role in the secure multiparty computation protocol of [CGMA85] as well as many important applications (like contract bidding, coin flipping, and electronic voting schemes, as exemplified in [CR87, DDN01, Gen00]) where broadcast is used in a more or less direct way.
The concept of simultaneous broadcast (also called independent broadcast) was first put forward by Chor et al. [CGMA85] who proposed a simulation-based definition, and presented protocols that securely implement simultaneous broadcast on top of a network which allows regular broadcast transmission operations, not necessarily satisfying the simultaneity property. The protocols in [CGMA85] require (for each simultaneous broadcast operation) a number of rounds that is linear in the number of parties. Given the importance of the simultaneous broadcast primitive, subsequent research efforts [CR87, Gen00] focused on reducing the round complexity, obtaining simultaneous broadcast protocols that run in logarithmically many [CR87] or even constant [Gen00] number of rounds (the latter result achieved in the common random string model.) Unfortunately, a close inspection of [CGMA85, CR87, Gen00] reveals that the definitions of simultaneous broadcast used in the three papers are quite different. Although, at first sight, all three definitions may appear appealing and intuitive, the technical differences among them bring up the following questions: what is the relation between the different definitions? Are they equivalent? Are they increasingly stronger or weaker? Or are they perhaps incomparable, in the sense that no one implies the other? Motivated by the efficiency improvement achieved by [CR87, Gen00] over the original linear round protocol of [CGMA85], we investigate and compare the definitions proposed in these three papers. (More precisely, we compare their straightforward generalizations to arbitrary input distributions[1].) Informally, our findings rank the original definition [CGMA85] as the strongest, and the most recent definition [Gen00] as the weakest. Technically, we prove implications and separations showing that the original definition [CGMA85] is strictly stronger (in a precise sense to be defined) than the definition of [CR87], which, in turn, is strictly stronger than the latest definition of [Gen00]. The comparison is not so straightforward because not all definitions are achievable for any input distribution, and for any pair of definitions (say, definition A and B) it may be possible to find a protocol Π and a distribution D such that Π is satisfies definition A but not definition B on

[1] At the time the definitions were suggested, a prime application of simultaneous broadcast was distributed coin flipping. Apparently influenced by that, the definitions of [CR87, Gen00] were implicitly understood to be used with uniform input distributions even though no such restriction was stated on the original papers.

input drawn according to D. So, it may seem that the definitions are incomparable. In order to properly rank the definitions, we first characterize the class of achievable input distributions for each definition. Our characterization is tight: for each definition A, we give a class of distributions $\mathcal{D}(A)$ such that

- definition A can be achieved in a strong sense: there exists a single protocol Π that satisfies A for any distribution in $\mathcal{D}(A)$
- the class $\mathcal{D}(A)$ cannot be extended even in a weak sense: for any distribution outside $\mathcal{D}(A)$, no protocol can possibly satisfy definition A.

It turns out that the class of distributions associated to the three definitions form a monotonically decreasing sequence. Let Sb, CR and G stand for the definitions given in [CGMA85], [CR87] and [Gen00] respectively, and let $\mathcal{D}(\mathsf{Sb})$, $\mathcal{D}(\mathsf{CR})$ and $\mathcal{D}(\mathsf{G})$ be the corresponding classes of input distributions. We show that

$$\mathcal{D}(\mathsf{Sb}) \supset \mathcal{D}(\mathsf{CR}) \supset \mathcal{D}(\mathsf{G}).$$

Armed with this characterization of the input distributions associated to each definition, we prove implications and separations between the three definitions as follows.
We prove that definition Sb implies definition CR in the sense that for any protocol Π, if Π is Sb-Independent for every distribution $D \in \mathcal{D}(\mathsf{CR})$ (i.e., for any distribution for which definition CR is achievable at all), then Π is also CR-Independent for every such distribution. Moreover, we give a simple example showing the reverse implication does not hold true, i.e., there exists a class of input distributions (such that Sb-Independence is achievable) and a protocol Π such that Π is CR-Independent but not Sb-Independent for every distribution in that class. We conclude that CR-independence is strictly weaker than Sb-Independence.
Next we prove that definition CR implies definition G in the sense that for any protocol Π, if Π is CR-Independent for any distribution $D \in \mathcal{D}(\mathsf{G})$, then Π is also G independent for any such input distribution. Moreover, we prove that the reverse implication is not true, i.e., there is a protocol Π that satisfies G-Independence for any distribution in $\mathcal{D}(\mathsf{G})$, but it does not satisfy CR-Independence for any nontrivial distribution (including the uniform). We conclude that G-independence is strictly weaker than CR-Independence.
We remark that while the relation between Sb-Independence and CR-Independence was to be expected because Sb resorts to a general secure multiparty computation definitional framework, the relation between CR-Independence and G-Independence was not as clear. In particular, [Gen00] seemed to suggest that the use of statistical notion of independence makes definition G stronger than CR, which uses a computational notion of closeness between distributions. Our results show that when restricted to an appropriate class of distributions, the relation between the two definitions is opposite to the one suggested in [Gen00].
We also remark that while simulation-based definitions are usually stronger than other definitions, and in many other cases in cryptography definitions have been made stronger and stronger over time, to culminate with a definition based on the simulation paradigm, the simultaneous broadcast problem studied in this paper represents an interesting case in which the reverse process has occurred: the original and strong simulation-based definition has been made weaker and weaker over time in order to achieve greater efficiency. We leave it as an open problem to find a constant-round protocol (i.e., as efficient as the one of [Gen00]) for simultaneous broadcast that achieves the stronger notion of CR-Independence [CR87] or even (and preferably) Sb-Independence [CGMA85].

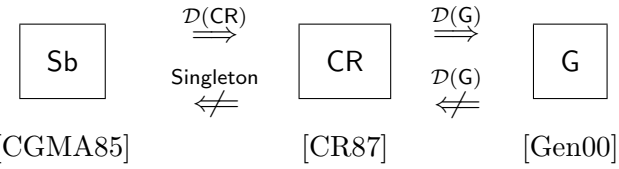

Figure 1: Our results. An arrow $\overset{\Delta}{\Longrightarrow}$ from definition A to B means that any protocol that achieves definition A under all distributions in Δ also achieves definition B under the same distributions. A broken arrow $\overset{\Delta}{\not\Longrightarrow}$ from A to B indicates that the implication $A \overset{\Delta}{\Longrightarrow} B$ is false.

Using Arbitrary Input Distributions: The question of whether security can be achieved under input distributions other than the uniform is not only of theoretical interest (comparing definitions) but of very practical relevance. In many applications (like electronic voting or contract bidding), the parties' input are not necessarily uniform or independent from each other – some partial knowledge of the inputs may have leaked. More general input distributions allow us to capture these cases. As a consequence, whether or not a definition of security can be achieved under more general input distributions can determine whether or not a given solution suffices for a particular application (e.g. whether the protocols suggested in [CR87, Gen00] guarantee security in scenarios with partial knowledge of the inputs, like voting). Given that the original definitions in [CR87, Gen00] did not explicitly excluded non-uniform input distributions, we see this contribution as useful in practice. Our characterization of the distributions associated to the definitions of [CR87] and [Gen00] show that those definitions are of limited applicability, as they can be achieved only for a restricted class of input distributions.

1.1 Related Work

In [DDN01], Dolev et al. introduce the notion of malleability of protocols, and present definitions for non-malleable message encryption, string commitment and zero-knowledge proofs. Loosely speaking, a protocol run by honest party P on private input x is *non-malleable* if no corrupted player P' can use (transform) the execution of the protocol to generate a valid execution of the same protocol under some input x' related to x. Therefore, non-malleability does guarantee some form of independence of the private values used in different protocols. The results of [DDN01], however, focus mostly on two-party protocols so their definitions do not capture the subtleties underlying the definition of independence of parallel broadcast protocols with more than two players. Along the same line, also in the two party setting, Liskov et al. [LLM$^+$01] study *mutually independent commitments* whose goal is to ensure the "independence" of the committed values. They give definitions which seem to capture – in a strong sense – this property. Their definitions, however, do not immediately extend to the multiparty case.

Organization: The paper is organized as follows. Section 2 presents some notation and terminology, and Section 3 describes the system model, including the definition of parallel broadcast. In Section 4, we present the definitions of independence existing in the literature, and in Section 5, a characterization of the sensible input distributions that can be associated to the definitions is made. Then, Section 6 presents implications and separations between the notions, and Section 7 concludes with some open problems.

2 Preliminaries

NOTATION: Let n be a positive integer, and $[n]$ denote the set $\{1, \ldots, n\}$. For any set $S \subset [n]$ and any vector $\mathbf{x} = (x_1, \ldots, x_n)$, we denote by $\mathbf{x}_S$ the $|S|$-dimensional vector formed by the elements of $\mathbf{x}$ whose index are in S, that is, $\mathbf{x}_S = (x_i)_{i \in S}$. Also, let G and B two disjoint sets such that $G \cup B = [n]$ and $\mathbf{w}, \mathbf{z}$ two n-dimensional vectors. Then, we let $\mathbf{w}_G \sqcup \mathbf{z}_B$ denote the n-dimensional vector formed by combining the elements of $\mathbf{w}$ with indexes in G with the elements of $\mathbf{z}$ with indexes in B. When clear from context, we may drop the subindex G or B, as in $\mathbf{w}_G \sqcup \mathbf{z}$. In such case, by convention, we assume the coordinates for $\mathbf{z}$ are in the set $\overline{G} = [n] \setminus G$.

PROBABILITY DISTRIBUTIONS, ENSEMBLES AND CLASSES OF DISTRIBUTIONS: A probability distribution $\mathcal{D}$ is a function from strings to non-negative reals such that $\sum_{x \in \{0,1\}^*} \mathcal{D}(x) = 1$. For any distribution $\mathcal{D}$ over $\{0,1\}^n$ we write $\mathbf{d} \xleftarrow{R} \mathcal{D}$ to denote the process of selecting an n-dimensional vector $\mathbf{d}$ from $\{0,1\}^n$ according to distribution $\mathcal{D}$. We also denote by $\mathcal{D}_B$, for any $B \subset [n]$, the distribution induced by selecting a vector in $\mathcal{D}$ and taking only the coordinates in set B. For simplicity, we write $\mathcal{D}_i$ instead of $\mathcal{D}_{\{i\}}$. We also extend the $\sqcup$ notation to distributions. Given two distributions $\mathcal{D}$ and $\mathcal{R}$ over n-bit strings, for any set $B \subset [n]$, we say an n-bit vector $\mathbf{x}$ is drawn from distribution $\mathcal{D}_B \sqcup \mathcal{R}_{\overline{B}}$ if $\mathbf{x}$ is formed by first drawing $\mathbf{x}_B$ from $\mathcal{D}_B$ and then drawing $\mathbf{x}_{\overline{B}}$ from $\mathcal{R}_{\overline{B}}$. Notice that for any distribution $\mathcal{D}$ and set B, $\mathcal{X} \stackrel{\text{def}}{=} \mathcal{D}_B \sqcup \mathcal{D}_{\overline{B}}$ is not necessarily equal to $\mathcal{D}$ since $\mathcal{X}_B$ is independent from $\mathcal{X}_{\overline{B}}$ while $\mathcal{D}_B$ and $\mathcal{D}_{\overline{B}}$ may be dependent.
A probability ensemble indexed by $\mathbf{N}$ is a sequence $\Delta = \{\mathcal{D}^{(k)}\}_{k \in \mathbf{N}}$ of probability distributions. For each value of the security parameter k, probability distribution $\mathcal{D}^{(k)}$ assigns positive probability only to n-bit strings. We sometimes abuse notation by using $\mathcal{D}^{(k)}$ to refer to the *random variable* that ranges over $\{0,1\}^n$ and that follows the corresponding distribution $\mathcal{D}^{(k)}$. As with distributions, given a probability ensemble $\mathcal{D} = \{\mathcal{D}^{(k)}\}$ and a set $B \subset [n]$, we let $\mathcal{D}_B = \{\mathcal{D}_B^{(k)}\}_{k \in \mathbf{N}}$ denote the ensemble consisting of the induced distributions $\mathcal{D}_B^{(k)}$. A class of probability ensembles (or simply, class of distributions) $\Phi = \{\Delta^{\langle \ell \rangle}\}_{\ell \in \mathbb{D}}$ is a collection of probability ensembles $\Delta^{\langle \ell \rangle}$ indexed by some (possibly uncountable) set $\mathbb{D}$.

ALGORITHMS AND THEIR PROBABILITIES: For any (probabilistic) algorithm A, $A(x)$ denotes the probability distribution of all possible outputs of running algorithm A on input x. If P is a predicate, A, B are (probabilistic) algorithms, and x, y are values, then $\Pr[\, a \leftarrow A(x),\ \ b \leftarrow B(y), \ldots : P(a, b, \ldots)\,]$ denotes the probability that predicate P on input $a, b, \ldots$ is true given that $a, b, \ldots$, are the output of the ordered execution of algorithm A on input x, B on input y, and so on. A function $\mu(k)$ is negligible in the security parameter k if there exists a constant $c > 0$ and infinitely many positive values of k such that $\mu(k) < k^{-c}$. A probability is overwhelming if it is larger than $1 - \mu(k)$ where $\mu(k)$ is a negligible function.

3 Parallel Broadcast

In this section, we describe some of the basic elements used in this work. We first describe the network model and then we formalize the concept of parallel broadcast.

3.1 The Model

We consider a network of n probabilistic, polynomial-time (PPT) parties (also called players) $P_1, P_2, \ldots, P_n$, where $n \in \mathbf{N}$ is some fixed constant. Each pair of players is connected by a point-to-point communication channel. We assume there is a probabilistic, polynomial-time adversary A

that statically corrupts some fixed fraction of the players (say, up to t of them) and is able to read all communication channels. The network is partially synchronous, which means parties have perfectly synchronized clocks which "tick" at discrete instants. The time interval between the i-th tick and the $(i+1)$-th tick is called the i-th round. Messages sent in one round are guaranteed to be delivered in the next round. The adversary is allowed *rushing*, which means that the network delivers the messages addressed to corrupted players instantly, so the adversary obtains those messages before deciding and sending out the messages of corrupted players for the same round. A protocol in this network is the collection of programs executed by these players.
We remark that our choice of network and adversary model is made mostly to fix ideas, since the model is rather orthogonal to the main focus of the paper, the definition of independence. Towards this end, we formalize the notion of parallel broadcast in the next section.

3.2 Parallel Broadcast

Intuitively, a parallel broadcast protocol is a broadcast protocol that allows all parties to broadcast values at the same time. Notice that, here, the term "parallel" refers to the property that multiple broadcast senders are allowed in the same protocol execution. The simplest instantiation of a parallel broadcast protocol is the protocol that performs n sequential executions of a standard (single-sender) broadcast protocol, where in the i-th execution party P_i acts as the sender.
Formally, assume each player P_i has an input bit x_i, and a security parameter k. (Henceforth, for simplicity, we consider the broadcast messages as bits). Consider a protocol Π run by the parties, at the end of which each honest party P_i outputs an n-dimensional vector $\mathbf{B}_i = (B_{i,1}, B_{i,2}, \ldots, B_{i,n}) \in \{0,1\}^n$. Protocol Π is said to implement *parallel broadcast* if it satisfies the following two properties:

(1) **Consistency**: For any adversary A, every honest parties P_i and P_j, $\mathbf{B}_i = \mathbf{B}_j$ with overwhelming probability.
(2) **Correctness**: For any adversary A, every honest parties P_i and P_j, $B_{i,j} = x_j$ with overwhelming probability.

The notion of parallel broadcast was introduced by Pease et al. in [PSL80] where it was called *interactive consistency*.
For every protocol that implements parallel broadcast it is possible to associate a single value to each party as the value *announced* by the party.

Definition 3.1 Assume parties $P_1, \ldots, P_n$ run some parallel broadcast protocol Π on input vector $\mathbf{x}$ under some polynomial-time adversary A. Then, for each $i \in \{1, \ldots, n\}$, we define the value "announced" by party P_i as the i-th bit output by any honest party P_k, namely $W_i \stackrel{\text{def}}{=} B_{k,i}$.[2] By the consistency property, the n-dimensional vector $\mathbf{W} = (W_1, \ldots, W_n)$ is well-defined with overwhelming probability. For notational convenience, we let $\textsc{Announced}_A^{\Pi}(\mathbf{x})$ denote vector $\mathbf{W}$ "announced" by the parties after running protocol Π under adversary A on input $\mathbf{x} \in \{0,1\}^n$. Similarly, $\textsc{Announced}_A^{\Pi}(\mathcal{X})$ denotes the induced distribution on $\textsc{Announced}_A^{\Pi}(\mathbf{x})$ when $\mathbf{x}$ is chosen according to some distribution $\mathcal{X}$. ▌

We remark that a parallel broadcast protocol does not necessarily guarantees independence of any sort – the announced values can be correlated even if the inputs are not. For example, the simplest instantiation described before (where n single-sender broadcasts are executed sequentially) satisfies both consistency and correctness but breaks independence: a dishonest last sender P_n could

[2] By convention, if a corrupted party P contributes with an invalid input or no input at all, honest parties assign the default value 0 as the bit "announced" by P.

discard its own input and broadcast one of the values previously heard (say, the one broadcast by party P_i). In this case, the i-th and n-th entry in the vector of announced values will always be the equal, no matter the inputs. More sophisticated parallel protocols like the expected constant-round interactive consistency protocol of Ben-Or and El-Yaniv [BOEY03] do not guarantee independence either.

4 Simultaneous Broadcast: Notions of Independence

Informally, a protocol Π is said to implement *simultaneous broadcast* (SB) if Π implements parallel broadcast where the values announced are "independent" of each other. Intuitively, the independence property sought must guarantee that no group of corrupted parties may announce values which may somehow depend on the values announced by any subset of the uncorrupted parties. In this section, we review some of the notions of independence previously proposed in the literature.

4.1 Chor, Goldwasser, Micali and Awerbuch's definition

In their seminal paper [CGMA85], Chor et al. define simultaneous broadcast as a network property that can be *emulated* starting from a network which provides a broadcast channel. Loosely speaking, Chor et al. show how to build a "compiler" that transforms protocols in a simultaneous broadcast network into protocols in a regular (non-simultaneous) broadcast network such that whatever an adversary can do in the latter network, there exists some adversary that can do the same in the former network.

Extracting a Simulation-based definition: We adapt the definition of [CGMA85] to the framework of secure function evaluation of [Can00] as follows. The case in which the parties have access to a simultaneous broadcast network is cast as the "ideal" process of Canetti's framework [Can00]. There, all parties have access to a trusted third party which computes the function $f_{SB}(\mathbf{x}) = (\mathbf{x}, \ldots, \mathbf{x})$. In the notation of [Can00], we call this protocol $\mathsf{Ideal}(f_{SB})$. On the other hand, to capture a regular (non-broadcast) network, we consider a "real" process in which a protocol Π is executed in a partially synchronous network under adversary A. Here, $\textsc{Exec}_A^{\Pi}(k, z, \mathbf{x})$ denotes the $(n+1)$-dimensional vector formed by the output of adversary A and the parties after executing protocol Π in the real process with inputs z and $\mathbf{x}$ respectively, and $\textsc{Exec}_S^{\mathsf{Ideal}(f_{SB})}(k, z, \mathbf{x})$ denotes the corresponding vector of outputs after $\mathsf{Ideal}(f_{SB})$ is executed with ideal adversary S in the ideal process (see Appendix B for details). Independence is then captured by requiring that Π securely implements f_{SB} in the sense of [Can00]. Thus, we obtain the following definition

Definition 4.1 [Sb-Independence] Protocol Π achieves Sb-independence if for any PPT adversary A corrupting up to $t < n$ parties, there exists a PPT simulator S such that, the ensembles (indexed by $k \in \mathbf{N}$, $\mathbf{x} \in \{0,1\}^n$, and $z \in \{0,1\}^*$),

$$\begin{aligned} \textsc{Exec}_A^{\Pi} &\stackrel{\text{def}}{=} \left\{\textsc{Exec}_A^{\Pi}(k, z, \mathbf{x})\right\} \\ \textsc{Exec}_S^{\mathsf{Ideal}(f_{SB})} &\stackrel{\text{def}}{=} \left\{\textsc{Exec}_S^{\mathsf{Ideal}(f_{SB})}(k, z, \mathbf{x})\right\} \end{aligned}$$

are computationally indistinguishable. ∎

Using input distributions: We also consider an alternative simulation-based definition which explicitly involves input distributions. This new definition, described next, is called (All, Sb)-Independence and it is shown to be equivalent to Sb-Independence in Appendix A.1.

Definition 4.2 [(Δ, Sb)-Independence] Let Δ be a class of input distribution ensembles over n-bit strings. Protocol Π achieves (Δ, Sb)-independence if for any PPT adversary A corrupting up to $t < n$ parties, there exists a PPT simulator S such that for every distribution ensemble $\mathcal{D} \in \Delta$, the ensembles (indexed by $k \in \mathbf{N}$, and $z \in \{0,1\}^*$)

$$\text{XEXEC}_A^{\Pi} \stackrel{\text{def}}{=} \left\{ \mathbf{x} \stackrel{R}{\leftarrow} \mathcal{D}^{(k)} \ : \ \left(\mathbf{x},\ \text{EXEC}_A^{\Pi}(k, z, \mathbf{x})\right) \right\} \tag{1}$$

$$\text{XEXEC}_S^{\mathsf{Ideal}(f_{SB})} \stackrel{\text{def}}{=} \left\{ \mathbf{x} \stackrel{R}{\leftarrow} \mathcal{D}^{(k)} \ : \ \left(\mathbf{x},\ \text{EXEC}_S^{\mathsf{Ideal}(f_{SB})}(k, z, \mathbf{x})\right) \right\} \tag{2}$$

are computationally indistinguishable. In this case, we say Π is Sb-Independent under class Δ. If $\Delta = \mathsf{All}$, the class of all input distributions over n-bit strings, then we say Π achieves $(\mathsf{All}, \mathsf{Sb})$-Independence. ▌

4.2 Chor and Rabin's definition

Chor and Rabin [CR87] proposed another definition of independence. Intuitively, their definition seems to come from the following idea. Let A be an adversary not corrupting party P_i. Any computable information on the $n-1$ bits announced by any party other than P_i can be cast as a (polynomial-time) predicate R on those bits. After fixing the adversary, whether or not this predicate is true defines an event. Then, if the bit output by P_i is probabilistically independent of any such event, then the output of P_i is effectively oblivious (unaffected) by the actions of adversary, thus guaranteeing some independence. A formal definition follows, slightly generalized to consider input distributions. The definition of [CR87], which was presented in a different but equivalent formulation (see Section A.2), is obtained as a special case when the input distribution is uniform.

Definition 4.3 (CR-Independence) Let $\mathcal{D}$ be an input distribution over $\{0,1\}^n$. A protocol Π achieves CR-independence under input distribution $\mathcal{D}$ if, for any adversary A, all honest party P_i, all polynomial-time predicate R, the quantity

$$\left| \Pr\left[\, \mathbf{W}_i = 0 \,\right] \cdot \Pr\left[R(\mathbf{W}_{\overline{\{i\}}}) \right] - \Pr\left[\mathbf{W}_i = 0 \ \wedge \ R(\mathbf{W}_{\overline{\{i\}}}) \right] \right| \tag{3}$$

is negligible (in the security parameter k) when $\mathbf{W} \leftarrow \text{ANNOUNCED}_A^{\Pi}(\mathcal{D}^{(k)})$. ▌

4.3 Gennaro's definition

The third definition of independence considered here was presented by Gennaro in [Gen00].[3] Loosely speaking, a protocol achieves independence under this definition if the bit announced by each corrupted party is not correlated with the bits announced by all the honest parties. In [Gen00], it is (implicitly) assumed the inputs to the parties follow the uniform distribution. Below, we slightly generalize the definition of [Gen00] to consider arbitrary input distributions.

Definition 4.4 (G-Independence) Let $\mathcal{D}$ be an input distribution over $\{0,1\}^n$. A protocol Π achieves G-independence under input distribution $\mathcal{D}$ if, for all adversaries A corrupting a subset B of parties (where $|B| = t < n$), for each corrupted party P_i, for all bit $b_i \in \{0,1\}$, and for all vectors $\mathbf{r}, \mathbf{s} \in \{0,1\}^{n-t}$ that occur with non-zero probability as $\mathcal{D}_{\overline{B}}$, the quantity

$$\left| \Pr\left[\, \mathbf{W}_i = b_i \ \mid \ \mathbf{W}_{\overline{B}} = \mathbf{r} \,\right] - \Pr\left[\, \mathbf{W}_i = b_i \ \mid \ \mathbf{W}_{\overline{B}} = \mathbf{s} \,\right] \right| \tag{4}$$

is negligible (in the security parameter k) when $\mathbf{W} \leftarrow \text{ANNOUNCED}_A^{\Pi}(\mathcal{D}^{(k)})$. ▌

[3] A different definition was originally described in a preliminary version [Gen95]. Since such definition evolved into the one of [Gen00], we do not considered it in this work.

A RELATED, SIMPLER DEFINITION: The idea behind the definition of [Gen00] is that, the probability that a corrupted party P_i outputs a bit b_i in the probability space where honest parties end up outputting a vector $\mathbf{r}$ must be about the same for any vector $\mathbf{r}$. This approach may lead to technical difficulties when proving properties of the definition over arbitrary distributions, since the definition may involve conditioning over possibly negligible events. To overcome this problem, we define a related (and possibly stronger) definition which implies Definition 4.4. The new definition, called G^{**}-Independence, is presented and shown to imply G-Independence in Appendix A.3. The fact that this new definition implies G-Independence will suffice to show implications and separations with respect to the other notions considered in this work.

5 The Role of the Input Distributions

The original definition of [CGMA85], although informal, is based on a general simulation paradigm and is arguably the strongest: a simultaneous broadcast protocol is a protocol that securely computes a function $f(x_1, \ldots, x_n)$ that on input n values $x_1, \ldots, x_n$ (provided by the n protocol participants) returns to each player the vector $\mathbf{x} = (x_1, \ldots, x_n)$ containing all the input values. Part of the power of this definition comes from the fact that security is required for any fixed input $(x_1, \ldots, x_n)$. This allows to model arbitrary input probability distributions, partial information about the inputs, etc.

In contrast, the definitions proposed in [CR87, Gen00] consider a specific input distribution and are statistical in nature: motivated by coin flipping applications, the definitions of [CR87, Gen00] consider the execution of the protocol when the input values $x_1, \ldots, x_n$ are chosen independently and uniformly at random, and propose a formalization of the intuitive requirement that

- the value broadcast by any honest party is independent from all other broadcast values [CR87], or
- the value broadcast by any corrupted party is independent from the values broadcast by all honest parties [Gen00].

Moreover, the notion of independence used in [CR87] is computational (i.e., it is only required that no polynomial time observer can detect dependencies), while the notion considered in [Gen00] is information theoretic. Both definitions can be generalized to arbitrary input distributions, but the generalization immediately highlights the limitations of the definitions in [CR87, Gen00]: if the input values $x_1, \ldots, x_n$ are strongly correlated, then the desired (correct) output also need to be correlated, and no protocol can possibly achieve the definition. In other words, there are probability distributions for which no protocol can possibly achieve the definitions in [CR87, Gen00]. At the same time, there are trivial distributions (e.g., any singleton distribution that concentrates all probability on a single input vector) for which any protocol vacuously satisfies the definition of [CR87, Gen00]. In other words, there are distributions for which the definitions of [CR87, Gen00] are not meaningful.

In this section, we formalize this intuition and for each definition of independence, we identify the largest class of distributions under which the definition is "achievable". More precisely, for each notion of independence, we prove there is a class of distributions under which the definition of independence can be realized – there exist a protocol that achieves the notion under such a class – but whose complement is not achievable in a strong sense: no protocol achieves the notion even for a single distribution outside the class. For any definition $\mathsf{N} \in \{\mathsf{CR}, \mathsf{G}\}$, we say a protocol Π achieves (Δ, N)-independence if Π achieves N-independence under every distribution in class Δ. We start by describing the input distributions for CR-Independence in next section.

5.1 Distributions for CR-Independence

COMPUTATIONALLY INDEPENDENT DISTRIBUTIONS: Let $\mathcal{X} = \{\mathcal{X}^{(k)}\}_{k\in\mathbf{N}}$ be a distribution ensemble such that every distribution $\mathcal{X}^{(k)}$ is the product of n arbitrary but independent distributions $X_1, \ldots, X_n$ over $\{0,1\}$, that is, $\mathcal{X}^{(k)} = X_1 \times X_2 \times \cdots \times X_n$. Ensembles with distributions of this form are called *independent*. Let $\Phi_n = \{\mathcal{X}^{\langle\ell\rangle}\}_{\ell\in\mathbb{D}}$ be the class of all independent n-dimensional ensembles, indexed by some (possibly uncountable) set $\mathbb{D}$. Let $\Psi_{C,n}$ be the class that contains all distributions ensembles computationally close to some distribution ensemble in Φ_n, that is, for each $\mathcal{D} \in \Psi_{C,n}$ there exist a distribution ensemble $\mathcal{X}$ in Φ_n such that $\mathcal{D}$ is computationally close to $\mathcal{X}$. If $\mathcal{D} \in \Psi_{C,n}$ we say $\mathcal{D}$ is a *computationally independent* distribution ensemble. Note that the ensembles for the uniform and all singleton distributions are indeed independent.

ACHIEVING CR-INDEPENDENCE: It is possible to show that, if the input distributions are computationally independent then CR-independence can be achieved. The proof of this result is postponed until Section 6.2.

Claim 5.1 Under *the assumption that enhanced trapdoor permutations exist* (cf. [Gol01, Sec. C.1]), there exists a protocol that achieves $(\Psi_{C,n}, \mathsf{CR})$-independence. ▌

Conversely, unless the input distribution $\mathcal{D}$ is computationally independent, no protocol can achieve independence according to Definition 4.3.

Lemma 5.2 Let Π be any parallel broadcast protocol and let $\mathcal{D} \notin \Psi_{C,n}$ be an input distribution ensemble. Then, Π does not achieve CR-independence under input distribution $\mathcal{D}$. ▌

Proof of Lemma 5.2: The proof uses the distinguisher that comes from $\mathcal{D}$ not being computationally independent to build a polynomial-time computable relation R and an adversary A that breaks the CR-independence of any protocol which runs on input distribution $\mathcal{D}$. Details follow.

Suppose $n \geq 2$ (otherwise the result holds vacuously). Let $\mathcal{D}$ be the input distribution ensemble not in $\Psi_{C,n}$ and, for some arbitrary index $i \in [n]$, consider the induced ensembles $\mathcal{D}_i$ and $\mathcal{D}_{\overline{\{i\}}} \stackrel{\text{def}}{=} \mathcal{D}_{[n]\setminus\{i\}}$. (Observe that all distributions over 1-bit are independent and $\mathcal{D}_i$ must be in Φ_1.) For distribution $\mathcal{D}_{\overline{\{i\}}}$ there are two cases depending on whether $\mathcal{D}_{\overline{\{i\}}} \in \Psi_{C,n-1}$ or not.

Let's analyze the first case, where $\mathcal{D}_{\overline{\{i\}}} \in \Psi_{C,n-1}$. Since $\mathcal{D} \notin \Psi_{C,n}$, we know $\mathcal{D}$ is not computationally close to any distribution in Φ_n. This means that, for every distribution $\mathcal{X} \in \Phi$, there exists a probabilistic polynomial-time adversary T and a constant $c > 0$ such that (w.l.o.g.) $\Pr\left[T(\mathcal{D}^{(k)}) = 1\right] - \Pr\left[T(\mathcal{X}^{(k)}) = 1\right] > k^{-c}$ for infinitely many values of the security parameter k. Take $\mathcal{X} = \mathcal{X}' \sqcup \mathcal{D}_i$ where $\mathcal{X}'$ is the ensemble in Φ_{n-1} which is computationally close to $\mathcal{D}_{\overline{\{i\}}}$. We want to prove that, in this case, Π cannot be CR-independent under input distribution $\mathcal{D}$: There must exist an adversary A, a polynomial-time predicate R, and a bit b such that $\left|\Pr[W_i = b] \cdot \Pr\left[R(\mathbf{W}_{\overline{\{i\}}})\right] - \Pr\left[W_i = b \wedge R(\mathbf{W}_{\overline{\{i\}}})\right]\right|$ is non-negligible. Indeed, it suffices to consider the trivial adversary A, which corrupts no party, and the predicate R defined by the execution of adversary T under the randomness that gives the "best" distinguishing advantage. Fix some arbitrary party $i \in [n]$, bit $b \in \{0,1\}$ and security parameter k. In what follows, we write T_r to denote running machine T with randomness fixed to the (sufficiently long) string $r \in \{0,1\}^*$. Then, for all bit b, for all $Z_{\overline{\{i\}}} \in \{0,1\}^{n-1}$, we define a predicate R on $Z_{\overline{\{i\}}}$ as $R^b(Z_{\overline{\{i\}}}) \stackrel{\text{def}}{=} T_r(Z_{\overline{\{i\}}} \sqcup b)$, where r is the value of τ that maximizes the difference $\Pr\left[T_\tau(\mathcal{D}^{(k)}) = 1\right] - \Pr\left[T_\tau(\mathcal{X}^{(k)}) = 1\right]$.

In what follows, we prove that, independently of protocol Π, there is a bit b such that adversary A and relation R^b violate the CR-independence of Π under input distribution $\mathcal{D}$. For simplicity, let

D and X denote the random variables corresponding to to distributions $\mathcal{D}^{(k)}_{\overline{\{i\}}}$ and $\mathcal{X}^{(k)}_{\overline{\{i\}}}$ respectively. Then, for $b \in \{0,1\}$, we have

$$\begin{aligned} p_b &\stackrel{\text{def}}{=} \Pr\left[\, W_i = b \,\wedge\, R^b(W_{\overline{\{i\}}}) \,\right] - \Pr\left[\, W_i = b \,\right] \cdot \Pr\left[\, R^b(W_{\overline{\{i\}}}) \,\right] \\ &= \Pr\left[\, T_r(D \sqcup b) \mid \mathcal{D}_i = b \,\right] \cdot \Pr\left[\, \mathcal{D}_i = b \,\right] - \Pr\left[\, \mathcal{D}_i = b \,\right] \Pr\left[\, T_r(D \sqcup b) \,\right] \end{aligned}$$

and, therefore

$$p_0 + p_1 = \Pr\left[\, T_r(\mathcal{D}) \,\right] - \left(\Pr\left[\, D_i = 0 \,\right] \cdot \Pr\left[\, T_r(D \sqcup 0) \,\right] + \Pr\left[\, D_i = 1 \,\right] \cdot \Pr\left[\, T_r(D \sqcup 1) \,\right]\right) \quad (5)$$

Now, let us consider the probability that T outputs 1 under distribution $\mathcal{X} \in \Phi_n$. Since $\mathcal{X}$ is the product of independent distributions

$$\Pr\left[\, T_r(\mathcal{X}) \,\right] = \Pr\left[\, X_i = 0 \,\right] \cdot \Pr\left[\, T_r(X \sqcup 0) \,\right] + \Pr\left[\, X_i = 1 \,\right] \cdot \Pr\left[\, T_r(X \sqcup 1) \,\right] \quad (6)$$

Subtracting Equation (6) from Equation (5), and using that $\mathcal{X}_i = \mathcal{D}_i$ and triangular inequality we obtain

$$\begin{aligned} |p_0| + |p_1| \;\geq\; & k^{-c} + \Pr\left[\, \mathcal{D}_i = 0 \,\right] \cdot \left(\Pr\left[\, T_r(X \sqcup 0) \,\right] - \Pr\left[\, T_r(D \sqcup 0) \,\right]\right) \\ & + \Pr\left[\, \mathcal{D}_i = 1 \,\right] \cdot \left(\Pr\left[\, T_r(X \sqcup 1) \,\right] - \Pr\left[\, T_r(D \sqcup 1) \,\right]\right) \;\geq\; k^{-c'} \,. \end{aligned}$$

for some constant $c' > 0$. Notice that the last inequality above follows from $\mathcal{D}_{\overline{\{i\}}} \in \Psi_{C,n-1}$.

For the second case, $\mathcal{D}_{\overline{\{i\}}} \notin \Psi_{C,n-1}$, we can apply the above argument on distribution $\mathcal{D}_{\overline{\{i\}}}$ instead, and inductively on n if necessary. This is possible if we pick i so the resulting distribution $\mathcal{D}_{\overline{\{i\}}}$ is *not* computationally independent. (If that is not possible, the first case above applies and the result holds.) The base case occurs when a distribution of the form $\mathcal{D}_{j_1} \times \mathcal{D}_{j_2}$ is reached. Then, $\mathcal{D}_{j_1}$ is clearly computationally close to itself and the first case above applies. This concludes the proof. ∎

5.2 Distributions for G-Independence

Locally Independent distributions: We say distribution ensemble $\mathcal{D}$ is *locally independent* if for all subset $B \subset [n]$, all string $\mathbf{u} \in \{0,1\}^{|B|}$, and all string $\mathbf{w} \in \{0,1\}^{n-|B|}$ that occurs with non-zero probability as $\mathcal{D}_{\overline{B}}$, the quantity

$$\left|\Pr\left[\, \mathcal{D}^{(k)}_B = \mathbf{u} \mid \mathcal{D}^{(k)}_{\overline{B}} = \mathbf{w} \,\right] - \Pr\left[\, \mathcal{D}^{(k)}_B = \mathbf{u} \,\right]\right|$$

is negligible in the security parameter k. We denote by $\Psi_{L,n}$ the class of all locally independent distribution ensembles.

Achieving G-Independence: It is possible to show that G-independence can be achieved under locally independent inputs. Again, the proof of this result is postponed until Section 6.2.

Claim 5.3 Under *the assumption that enhanced trapdoor permutations exist* (cf. [Gol01, Sec. C.1]), there exists a protocol that achieves $(\Psi_{L,n}, \mathsf{G})$-independence. ∎

On the other hand, the following result shows that no protocol is G-independent under input distributions which are not locally independent.

Lemma 5.4 Let Π be any parallel broadcast protocol and $\mathcal{D} \notin \Psi_{L,n}$ be an input distribution. Then, Π does not achieve G-independence under input distribution $\mathcal{D}$. ▌

Proof of Lemma 5.4: Fix a parallel broadcast protocol Π. Let $\mathcal{D}$ be a distribution not in $\Psi_{L,n}$. Then, by definition, there exist a subset $B \subset [n]$, $|B| = t$, a string $\mathbf{u} \in \{0,1\}^t$, and a string $\mathbf{w} \in \{0,1\}^{n-t}$, such that

$$\left|\Pr\left[\mathcal{D}_B^{(k)} = \mathbf{u} \mid \mathcal{D}_{\overline{B}}^{(k)} = \mathbf{w}\right] - \Pr\left[\mathcal{D}_B^{(k)} = \mathbf{u}\right]\right|$$

is not negligible. To prove the result it suffices to choose an arbitrary index $i \in B$, and an adversary A that works as follows: A corrupts parties in B and announces 1 for (corrupted) party P_i only when $\mathbf{x}_B = \mathbf{u}$. Also, with overwhelming probability, since $\mathbf{x} \stackrel{R}{\leftarrow} \mathcal{D}^{(k)}$ we know that $\mathbf{W}_{\overline{B}} = \textsc{Announced}_A^{\Pi}(\mathbf{x})_{\overline{B}}$ follows distribution $\mathcal{D}_{\overline{B}}^{(k)}$. Then,

$$\begin{aligned} &\left|\Pr\left[\mathbf{W}_i = 1 \mid \mathbf{W}_{\overline{B}} = \mathbf{w}\right] - \Pr\left[\mathbf{W}_i = 1\right]\right| \\ = \;& \left|\Pr\left[\mathcal{D}_B^{(k)} = \mathbf{u} \mid \mathcal{D}_{\overline{B}}^{(k)} = \mathbf{w}\right] - \Pr\left[\mathcal{D}_B^{(k)} = \mathbf{u}\right]\right| \end{aligned}$$

is not negligible either. The result then follows by Proposition C.1. ▌

5.3 Distributions for Sb-Independence

In this section, we show that Sb-Independence can be achieved under any input distribution. We first notice that Sb-Independence under class $\mathsf{Singleton}$ and $(\mathsf{All}, \mathsf{Sb})$-Independence are equivalent (this follows from Proposition A.1). Then, we recall the results by Yao and (independently) by Goldreich et al. [Yao86, GMW87] which present protocols that securely implement any function. In particular, these protocols securely implement f_{SB}. By observing that such protocols work for any fixed input, we then have

Corollary 5.5 [Yao86, GMW87] Under *the assumption that enhanced trapdoor permutations exist* (cf. [Gol01, Sec. C.1]), there exists a protocol that achieves Sb-independence for any input distribution. ▌

5.4 Relations between Distributions

We introduce some notation first. Let $\mathsf{Singleton}$ be the class of all singleton input distribution ensembles. That is, for each string $\alpha \in \{0,1\}^n$, the distribution $\mathcal{D}_\alpha = \{\mathcal{D}_\alpha^{(k)}\}_{k \in \mathbf{N}}$ is in $\mathsf{Singleton}$ if for every k, $\mathcal{D}_\alpha^{(k)}$ assigns probability one to the string α. Let $\mathsf{Uniform}$ be the class whose only element is the uniform distribution ensemble, and let All be the class of all input distribution ensembles over n-bit strings. For notational convenience, in the rest of the paper, we denote by $\mathcal{D}(\mathsf{N})$ the class of distributions associated to definition N, that is, $\mathcal{D}(\mathsf{CR}) \stackrel{\text{def}}{=} \Phi_{C,n}$, $\mathcal{D}(\mathsf{G}) \stackrel{\text{def}}{=} \Phi_{L,n}$, and $\mathcal{D}(\mathsf{Sb}) \stackrel{\text{def}}{=} \mathsf{All}$. The following claim shows that the input distributions under which G, CR, and Sb are achievable are strictly contained in the same order. All classes also contain the class of all singleton distributions and the class of the uniform distribution. The proofs are easy and therefore omitted.

Claim 5.6 $\mathsf{Singleton}, \mathsf{Uniform} \subsetneq \mathcal{D}(\mathsf{G}) \subsetneq \mathcal{D}(\mathsf{CR}) \subsetneq \mathcal{D}(\mathsf{Sb})$. ▌

6 Implications and Separations

In this section, we compare the definitions of independence of [CR87, Gen00] with the simulation-based definition. We say a distribution is *trivial* for notion N if every protocol achieves N-independence under that input distribution. Also, a class is trivial for notion N if every protocol achieves N-Independence under all distributions in the class. Our first implication shows that any protocol that achieves Sb-Independence must achieve CR-Independence for all achievable distributions.

Lemma 6.1 For every protocol Π, if Π achieves $(\mathcal{D}(\mathsf{CR}), \mathsf{Sb})$-Independence then Π also achieves $(\mathcal{D}(\mathsf{CR}), \mathsf{CR})$-Independence. ∎

Proof of Lemma 6.1: Assume parallel broadcast protocol Π is not CR-independent for some input distribution $\mathcal{D} \in \mathcal{D}(\mathsf{CR})$. Then, there exists an adversary A, honest party P_ℓ, and a polynomial-time predicate R such that the quantity defined in Definition 4.3 is not negligible under input distribution $\mathcal{D}$. We show how to transform A, R and $\mathcal{D}$, into an adversary A', and an algorithm T such that Π is not $(\mathcal{D}(\mathsf{CR}), \mathsf{Sb})$-independent. Details follow.

First, since Π is not CR-independent under input distribution $\mathcal{D} \in \mathcal{D}(\mathsf{CR})$, there must exist an adversary A (corrupting players in $B \subset [n]$), an honest party P_i, and a polynomial-time computable predicate R such that there exists constant $c > 0$ and infinitely many values of k for which (w.l.o.g.)

$$\Pr\left[\mathbf{W}_\ell = 1 \ \wedge \ R(\mathbf{W}_{\overline{\{\ell\}}})\right] - \Pr\left[\mathbf{W}_\ell = 1\right] \cdot \Pr\left[R(\mathbf{W}_{\overline{\{\ell\}}})\right] \ \geq \ k^{-c} \tag{7}$$

Now, let adversary A' be identical to A. We build distinguisher T from predicate R as follows:

Distinguisher $T\,(1^k, z, \mathbf{x}, \tau)$: From transcript τ extract unique $\mathbf{W}$

Output 1 if $\left(\mathbf{W}_\ell = 1 \text{ and } R(\mathbf{W}_{\overline{\{\ell\}}}) = 1\right)$ and 0 otherwise.

Algorithm T is polynomial-time since R is so. It remains to prove that T successfully distinguishes ensembles $\text{XEXEC}^{\Pi}_{A'}$ and $\text{XEXEC}^{\mathsf{Ideal}(f_{SB})}_{S}$ (as defined in equations (1) and (2)) when the input distribution is $\mathcal{D}$.

Let S be an ideal process adversary (simulator). We denote by $S(\mathbf{x}_B; z)$ the $|B|$-dimensional vector given by simulator S to function f_{SB} (in the ideal world) as the input corresponding to corrupted parties. String z is the auxiliary input of S. Let $\Pr_1[E]$ be the probability of event E under the case $\mathbf{x} \stackrel{R}{\leftarrow} \mathcal{D}^{(k)}$ and $\mathbf{W} \leftarrow \text{ANNOUNCED}^{\Pi}_{A}(\mathbf{x})$, and $\Pr_0[E]$ be the probability of event E under the choice $\mathbf{x} \stackrel{R}{\leftarrow} \mathcal{D}^{(k)}$ and $\mathbf{W} \leftarrow \mathbf{x}_{\overline{B}} \sqcup S(\mathbf{x}_B)$. Then,[4]

$$p_1 \stackrel{\text{def}}{=} \Pr\left[\mathbf{x} \stackrel{R}{\leftarrow} \mathcal{D}^{(k)} \ : \ T(1^k, z, \mathbf{x}, \text{EXEC}^{\Pi}_{A'}(k, z, \mathbf{x})) = 1\right] = \Pr_1\left[\mathbf{W}_\ell = 1 \ \wedge \ R(\mathbf{W}_{\overline{\{\ell\}}}) = 1\right]$$

$$p_0 \stackrel{\text{def}}{=} \Pr\left[\mathbf{x} \stackrel{R}{\leftarrow} \mathcal{D}^{(k)} \ : \ T(1^k, z, \mathbf{x}, \text{EXEC}^{\mathsf{Ideal}(f_{SB})}_{S}(k, z, \mathbf{x})) = 1\right]$$

$$= \Pr_0\left[\mathbf{x}_\ell = 1 \ \wedge \ R(\mathbf{x}_{\overline{B}\setminus\{\ell\}} \sqcup S(\mathbf{x}_B; z)) = 1\right]$$

At this point, we use that $\mathcal{D}$ is computationally independent. Let $\mathcal{X} \stackrel{\text{def}}{=} \mathcal{D}_{\overline{\{\ell\}}} \sqcup \mathcal{D}_\ell$. By a hybrid argument, we assume $\mathcal{X} \in \Phi_n$. Then, there exists a negligible function $\epsilon(k)$ such that

[4] In the rest of the proofs in this paper, for simplicity, we assume that $\mathbf{W}_\ell = \mathbf{x}_\ell$ with probability one for all uncorrupted P_ℓ. The cases when the equality holds with overwhelming probability are analogous, although slightly more involved.

$|\Pr\left[\, F(\mathcal{D}^{(k)}) = 1 \,\right] - \Pr\left[\, F(\mathcal{X}^{(k)}) = 1 \,\right]| < \epsilon(k)$ for any probabilistic polynomial-time distinguisher F, in particular $F(\mathbf{Z}) \stackrel{\text{def}}{=} (\mathbf{Z}_\ell = 1 \wedge R(\mathbf{Z}_{\overline{B}\setminus\{\ell\}} \sqcup S(\mathbf{Z}_B; z)) = 1)$. Therefore,

$$\begin{aligned}
p_0 &< \Pr\left[\mathbf{u} \stackrel{R}{\leftarrow} \mathcal{X}^{(k)} \,:\, \mathbf{u}_\ell = 1 \wedge R(\mathbf{u}_{\overline{B}\setminus\{\ell\}} \sqcup S(\mathbf{u}_B; z)) = 1 \right] + \epsilon(k) \\
&= \Pr\left[\mathbf{u}_{\overline{\{\ell\}}} \stackrel{R}{\leftarrow} \mathcal{D}_B^{(k)} \,:\, R(\mathbf{u}_{\overline{B}\setminus\{\ell\}} \sqcup S(\mathbf{u}_B; z)) = 1 \mid \mathbf{u}_\ell = 1 \right] \cdot \Pr\left[\mathbf{u}_\ell \stackrel{R}{\leftarrow} \mathcal{D}_\ell^{(k)} \,:\, \mathbf{u}_\ell = 1 \right] + \epsilon(k) \\
&= \Pr_0\left[R(\mathbf{x}_{\overline{B}\setminus\{\ell\}} \sqcup S(\mathbf{x}_B; z)) = 1 \right] \cdot \Pr_0\left[\, \mathbf{W}_\ell = 1 \,\right] + \epsilon(k) \\
&< \Pr_1\left[R(\mathbf{W}_{\overline{\{\ell\}}}) \right] \cdot \Pr_0\left[\, \mathbf{W}_\ell = 1 \,\right] + \epsilon(k)
\end{aligned}$$

We justify last inequality as follows: (a) if $\Pr_1\left[R(\mathbf{W}_{\overline{\{\ell\}}}) \right] < \Pr_0\left[R(\mathbf{W}_{\overline{\{\ell\}}}) \right]$ then it suffices to consider the negated predicate $\overline{R}$ instead of R, and (b) any adversary A cannot use the simulator S's strategy otherwise A would contradict Equation (7) since $\mathcal{D} \in \mathcal{D}(\mathsf{CR})$. Also, by the correctness of Π, $\mathbf{W}_i = \mathbf{x}_i$ for all honest $i \in \overline{B}$. Combining the above equations with Equation (7), we obtain

$$p_1 - p_0 \;>\; \Pr_1\left[\mathbf{W}_\ell = 1 \wedge R(\mathbf{W}_{\overline{\{\ell\}}}) = 1 \right] - \Pr_1\left[R(\mathbf{W}_{\overline{\{\ell\}}}) \right] \cdot \Pr_1\left[\, \mathbf{W}_\ell = 1 \,\right] - \epsilon(k) \;>\; k^{-c'}$$

for some constant $c' > 0$ and infinitely many values of k. ∎

Similarly, all protocols that achieve CR-Independence under all distributions for which G-Independence is achievable must indeed achieve G-Independence under the same class.

Lemma 6.2 For every protocol Π, if Π achieves $(\mathcal{D}(\mathsf{G}), \mathsf{CR})$-Independence then Π also achieves $(\mathcal{D}(\mathsf{G}), \mathsf{G})$-Independence. ∎

Proof of Lemma 6.2: Let Π be a parallel broadcast protocol. Assume Π is not G-Independent under some distribution $\mathcal{D}$. We want to prove that there exist a distribution $\mathcal{D}'$ under which Π is not CR-Independent. By Proposition A.7, if Π does not achieve G-Independence under distribution $\mathcal{D}$, then Π is not G^{**}-Independent. Therefore, there exists an polynomial-time adversary A corrupting set $B \subset [n]$, a string $z \in \{0,1\}^*$, $i \in B$, and vectors $\mathbf{w} \in \{0,1\}^B$, $\mathbf{r}, \mathbf{s} \in \{0,1\}^{\overline{B}}$ such that the quantity

$$\left| \Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{w} \sqcup \mathbf{r}) \,:\, \mathbf{W}_i = 1 \right] - \Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{w} \sqcup \mathbf{s}) \,:\, \mathbf{W}_i = 1 \right] \right|$$

is not negligible. By a hybrid argument, we can assume $\mathbf{r}$ and $\mathbf{s}$ differ on a single bit, the ℓ-th one, so $\mathbf{r}_{B\setminus\{\ell\}} = \mathbf{s}_{B\setminus\{\ell\}}$. W.l.o.g. $\mathbf{r}_\ell = 0$ and $\mathbf{s}_\ell = 1$.

We build a new adversary A' identical to A and fix the honest player P_ℓ. We also define the predicate $R(\mathbf{Z}_{\overline{\ell}}) \stackrel{\text{def}}{=} (\mathbf{Z}_i \stackrel{?}{=} 1)$. Now, consider the distribution $\mathcal{D}'$ that assigns some non-negligible probability p_ℓ to the event $\mathcal{D}'^{(k)}_\ell = 1$, and probability one to $\mathcal{D}'^{(k)}_{\overline{\{\ell\}}} = (\mathbf{w} \sqcup \mathbf{r}_{\overline{B}\setminus\{\ell\}})$. Notice that $\mathcal{D}'^{(k)}$ is in $\mathcal{D}(\mathsf{G})$ but it is not trivial. Let $\Pr_{\mathcal{D}'}\left[\, E \,\right]$ the probability of event E when $\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathcal{D}'^{(k)})$. Since P_ℓ is honest $\Pr_{\mathcal{D}'}\left[\, \mathbf{W}_\ell = 1 \,\right] = \Pr\left[\mathcal{D}'^{(k)}_\ell = 1 \right] = p_\ell$. Then,

$$\begin{aligned}
\Pr_{\mathcal{D}'}\left[R(\mathbf{W}_{\overline{\{\ell\}}}) = 1 \right] &= (1 - p_\ell) \cdot \Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{w} \sqcup \mathbf{r}) \,:\, \mathbf{W}_i = 1 \right] \\
&\quad + p_\ell \cdot \Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{w} \sqcup \mathbf{s}) \,:\, \mathbf{W}_i = 1 \right]
\end{aligned}$$

and

$$\begin{aligned}\Pr_{\mathcal{D}'}\left[\mathbf{W}_\ell = 1 \wedge R(\mathbf{W}_{\overline{\{\ell\}}}) = 1\right] &= \Pr_{\mathcal{D}'}\left[\mathbf{W}_\ell = 1 \wedge \mathbf{W}_i = 1\right] \\ &= p_\ell \cdot \Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{w} \sqcup \mathbf{s}) \,:\, \mathbf{W}_i = 1\right]\end{aligned}$$

Putting it all together,

$$\begin{aligned}&\left|\Pr_{\mathcal{D}'}\left[\mathbf{W}_\ell = 1\right] \cdot \Pr_{\mathcal{D}'}\left[R(\mathbf{W}_{\overline{\{\ell\}}}) = 1\right] - \Pr_{\mathcal{D}'}\left[\mathbf{W}_\ell = 1 \wedge R(\mathbf{W}_{\overline{\{\ell\}}}) = 1\right]\right| \\ &= p_\ell \cdot (1 - p_\ell) \cdot \left|\Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{w} \sqcup \mathbf{r}) \,:\, \mathbf{W}_i = 1\right]\right. \\ &\qquad \left. - \Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{w} \sqcup \mathbf{s}) \,:\, \mathbf{W}_i = 1\right]\right|\end{aligned}$$

which is not negligible. ▌

6.1 Separations

At this point, we look into whether the definitions are equivalent when restricted to achievable input distributions. Proposition 6.3 shows this is not the case. We prove that there are distributions for which the definition of [CR87] always holds no matter the protocol, but that this cannot happen with Sb-Independence.

Proposition 6.3 The class Singleton is trivial for CR independence but not trivial for Sb independence. ▌

Proof of Proposition 6.3: The proof follows easily from the definition of CR independence – under a fixed input all probabilities collapse to either 0 or 1 with overwhelming probability, for any protocol. Then, consider a protocol that does not achieve Sb-Independence (we know such protocol exist). Since (Singleton, Sb)-Independence is equivalent to Sb-Independence, it follows that such protocol cannot achieve Sb-Independence under class Singleton. ▌

It is also possible to show that the definitions of [CR87] and [Gen00] are not equivalent, but instead that G-independence is strictly weaker than CR-independence.

Lemma 6.4 There exists a protocol Π_G which achieves $(\mathcal{D}(\mathsf{G}), \mathsf{G})$-independence but does not achieve CR-independence for any input distribution in $\mathcal{D}(\mathsf{G})$. In particular, Π_G is G-Independent for the uniform distribution, but not CR-Independent for the uniform distribution. ▌

Proof: We show a protocol implementing parallel broadcast that, even though it satisfies Definition 4.4 (i.e., the notion of simultaneous broadcast of [Gen00]), it violates Definition 4.3 (i.e., the definition of independence of [CR87]). The "flawed" protocol Π_G uses a subprotocol Θ which essentially performs a simultaneous broadcast *unless* two corrupted parties misbehave in a very controlled manner – by setting some auxiliary input bit to 1. In such case, protocol Θ reveals some information about the honest parties' inputs to two corrupted parties. The leakage of information is done in such a way that the output of each single corrupted party is *not* correlated to the outputs of honest parties, but the *combined* outputs are.

We describe protocol Θ first. Protocol Θ is a n-party protocol that securely implements function $g(\mathbf{v})$ on input $\mathbf{v} = (v_1, \ldots, v_n)$ defined as

$$g(\mathbf{v}) \stackrel{\text{def}}{=} \left\{ \begin{array}{l} \text{First, parse each } v_i \text{ as } (x_i, b_i) \\ \text{Pick } r \stackrel{R}{\leftarrow} \{0,1\} \text{ and set } \mathcal{L} \leftarrow \{\, i \,:\, b_i = 1 \,\} \\ \text{If } |\mathcal{L}| = 2 \text{ then set } \ell_1, \ell_2 \in \mathcal{L},\ \ell_1 < \ell_2, \text{ otherwise set } \ell_1, \ell_2 \leftarrow 0 \\ \text{Compute } y \stackrel{R}{\leftarrow} \bigoplus_{i \notin \{\ell_1, \ell_2\}} x_i \\ \text{Set } w_i \leftarrow \left\{ \begin{array}{ll} r & \text{if } |\mathcal{L}| = 2 \text{ and } i = \ell_1 \\ r \oplus y & \text{if } |\mathcal{L}| = 2 \text{ and } i = \ell_2 \\ x_i & \text{if } i \neq \ell_1, \ell_2 \end{array} \right. \\ \text{Set } \mathbf{w} \leftarrow (w_1, \ldots, w_n) \text{ and output the } n\text{-dimensional vector } (\mathbf{w}, \mathbf{w}, \ldots, \mathbf{w}) \end{array} \right.$$

For simplicity, we write the input vector $\mathbf{v}$ as $\mathbf{v} = (\mathbf{x}, \mathbf{b})$, where $\mathbf{x}, \mathbf{b} \in \{0,1\}^n$. We first notice that a protocol that securely implements function g can be built using known techniques (cf. [BGW88, GMW87, CCD88]) as long as $t < \lceil n/2 \rceil$.

Claim 6.5 There exist a protocol Θ that securely implements g (in the sense of [Can00]). ▌

We now describe protocol Π_G. On private input $x_i \in \{0,1\}$, each party P_i sets up an auxiliary bit $b_i \leftarrow 0$. Then, all parties call subprotocol Θ on input $((x_1, b_1), (x_2, b_2), \ldots, (x_n, b_n))$. Let $\mathbf{W}_i$ be the vector obtained as the output of protocol Θ by party P_i. Each party P_i outputs $\mathbf{W}_i$ as the final protocol result.

We show that protocol Π_G is *not* CR-Independent under any non-trivial input distribution. Indeed, there exists an adversary A^* such that, when protocol Π_G is executed on any input $\mathbf{x}$ under adversary A^*, the sum (mod 2) of the announced bits is always zero. Adversary A^* corrupts only two parties and instructs them to set their auxiliary bits to 1. The next claim follows directly from the definition of g.

Claim 6.6 Assume parties have inputs chosen according to some arbitrary distribution $\mathcal{D} \in \mathcal{D}(\mathsf{G})$. There exists an adversary A^* such that the execution of protocol Π_G on input $\mathbf{x} \in \mathcal{D}$ under adversary A^* defines a vector of announced bits $\mathbf{W}$ satisfying $\bigoplus_i W_i = 0$. ▌

The attack works for any non-trivial distribution, i.e., any distribution that is not statistically close to a singleton. For any such distribution, there must exists and index i such that $1/poly < \Pr[\,\mathbf{W}_i = 0\,] < 1 - 1/poly$. The above claim gives an adversary and a polynomial-time predicate we can use to correlate the output of the corrupted parties with the output of an honest party P_i, namely $R(\mathbf{Z}_{\overline{\{i\}}}) \stackrel{\text{def}}{=} (\oplus_{j \neq i} \mathbf{Z}_j = 0)$. Notice that the predicate holds if and only if P_i announces 0.

We now show that protocol Π_G achieves G-Independence for any non-trivial, locally independent input distribution $\mathcal{D}$. Indeed, for any adversary A that succeeds on attacking the G-Independence of Π_G under $\mathcal{D}$, we exhibit a distinguisher Q that contradicts the security of Θ (Claim 6.5). We proceed as follows. Assume Π_G is not G-Independent. By Proposition A.7, Π_G is not G^{**}-Independent. Then there is an adversary A which corrupts parties in B, an auxiliary input τ, and a corrupted party P_i, for which there are vectors $\mathbf{w} \in \{0,1\}^B$, $\mathbf{r}, \mathbf{s} \in \{0,1\}^{\overline{B}}$, such that

$$\Big| \Pr\Big[\, \mathbf{W} \leftarrow \textsc{Announced}^{\Pi_G}_{A(k,\tau)}(\mathbf{w} \sqcup \mathbf{r}) \,:\, \mathbf{W}_i = 1 \,\Big] - \Pr\Big[\, \mathbf{W} \leftarrow \textsc{Announced}^{\Pi_G}_{A(k,\tau)}(\mathbf{w} \sqcup \mathbf{s}) \,:\, \mathbf{W}_i = 1 \,\Big] \Big|$$

is not negligible. By a hybrid argument, we can assume $\mathbf{r}$ and $\mathbf{s}$ differ in a single bit, the ℓ-bit, so $r_\ell \neq s_\ell$, and w.l.o.g, $r_\ell = 0$ and $s_\ell = 1$.

The above adversary gives us a procedure to guess the input bit used by honest party P_ℓ in protocol Θ as long as the inputs vector for the remaining parties is equal to $\mathbf{w} \sqcup \mathbf{r}_{\overline{B}\setminus\{\ell\}}$. Indeed, starting from A we show how to build an adversary A' for Θ, such that for any ideal-process adversary S for $\mathsf{Ideal}(g)$, there exist a distinguisher Q, an auxiliary input z', an input vector $\mathbf{v}' = (\mathbf{x}', \mathbf{b}')$, such that the quantity

$$\left|\Pr\left[\, Q(1^k, z', \mathbf{v}', \textsc{Exec}^{\Theta}_{A'}(k, z', \mathbf{v}')) = 1 \,\right] - \Pr\left[\, Q(1^k, z', \mathbf{v}', \textsc{Exec}^{\mathsf{Ideal}(g)}_{S}(k, z', \mathbf{v}')) = 1 \,\right]\right|$$

is not negligible. Adversary A' is simple. It corrupts the same parties as B, and works as follows. On input $(\mathbf{x}_B, \mathbf{b}_B)$, A' simply discards $\mathbf{b}_B$ and then simulates A. We now set $\mathbf{b}' = \mathbf{0}$ and $z' = \tau$. For simplicity, for any vector $\mathbf{x} \in \{0,1\}^n$, denote

$$\begin{aligned} q_{real,\mathbf{x}} &\stackrel{\text{def}}{=} \Pr\left[\, Q(1^k, z', (\mathbf{x}, \mathbf{b}'), \textsc{Exec}^{\Theta}_{A'}(k, z', (\mathbf{x}, \mathbf{b}'))) = 1 \,\right], \\ q_{ideal,\mathbf{x}} &\stackrel{\text{def}}{=} \Pr\left[\, Q(1^k, z', (\mathbf{x}, \mathbf{b}'), \textsc{Exec}^{\mathsf{Ideal}(g)}_{S}(k, z', (\mathbf{x}, \mathbf{b}'))) = 1 \,\right] \end{aligned}$$

It remains to show a distinguisher algorithm Q that works with good probability. Our algorithm Q takes as input a security parameter $k \in \mathbf{N}$, an auxiliary string $z \in \{0,1\}^*$, a vector $\mathbf{v} = (\mathbf{x}, \mathbf{b}) \in \{0,1\}^n \times \{0,1\}^n$, and a string Z drawn either from distribution $\textsc{Exec}^{\Theta}_{A'}(k, z, \mathbf{v})$ or distribution $\textsc{Exec}^{\mathsf{Ideal}(g)}_{S}(k, z, \mathbf{v})$. Thus, on input $(1^k, z, (\mathbf{x}, \mathbf{b}), Z)$, algorithm Q first extract the corrupted set B and the unique vector $\mathbf{W} = (W_1, \ldots, W_n)$ of announced values from transcript Z. Then, it simply outputs 1 if $(W_i = W_\ell)$, and 0 otherwise. Let $\mathbf{x}^{\mathbf{r}} = \mathbf{w} \sqcup \mathbf{r}$ and $\mathbf{x}^{\mathbf{s}} = \mathbf{w} \sqcup \mathbf{s}$. By definition of distinguisher Q and adversary A', in the real model we have that

$$\begin{aligned} q_{real,\mathbf{x}^{\mathbf{s}}} &= \Pr\left[\, \mathbf{W} \leftarrow \textsc{Announced}^{\Pi_G}_{A(k,\tau)}(\mathbf{x}^{\mathbf{s}}) \,:\, \mathbf{W}_i = 1 \,\right] \\ q_{real,\mathbf{x}^{\mathbf{r}}} &= \Pr\left[\, \mathbf{W} \leftarrow \textsc{Announced}^{\Pi_G}_{A(k,\tau)}(\mathbf{x}^{\mathbf{r}}) \,:\, \mathbf{W}_i = 0 \,\right] \end{aligned}$$

In the ideal model, on the other hand, the adversary S has access only to $\mathbf{x}_B = \mathbf{w}$, and therefore

$$q_{ideal,\mathbf{x}^{\mathbf{r}}} \;=\; 1 - \Pr\left[\, S(\mathbf{w}; \tau)_i = 1 \,\right] \quad \text{and} \quad q_{ideal,\mathbf{x}^{\mathbf{s}}} \;=\; \Pr\left[\, S(\mathbf{w}; \tau)_i = 1 \,\right]$$

Combining the above equations, we obtain

$$\begin{aligned} &|q_{real,\mathbf{x}^{\mathbf{s}}} - q_{ideal,\mathbf{x}^{\mathbf{s}}}| + |q_{real,\mathbf{x}^{\mathbf{r}}} - q_{ideal,\mathbf{x}^{\mathbf{r}}}| \;\geq\; |q_{real,\mathbf{x}^{\mathbf{s}}} - q_{real,\mathbf{x}^{\mathbf{r}}} - (q_{ideal,\mathbf{x}^{\mathbf{r}}} + q_{ideal,\mathbf{x}^{\mathbf{s}}})| \\ &= \left|\Pr\left[\, \mathbf{W} \leftarrow \textsc{Announced}^{\Pi_G}_{A(k,\tau)}(\mathbf{x}^{\mathbf{s}}) \,:\, \mathbf{W}_i = 1 \,\right] - \Pr\left[\, \mathbf{W} \leftarrow \textsc{Announced}^{\Pi_G}_{A(k,\tau)}(\mathbf{x}^{\mathbf{r}}) \,:\, \mathbf{W}_i = 1 \,\right]\right| \end{aligned}$$

which is not negligible by the G^{**}-Independence. Therefore, for either input $\mathbf{x}' = \mathbf{x}^{\mathbf{s}}$ or input $\mathbf{x}' = \mathbf{x}^{\mathbf{r}}$, the quantity $|q_{real,\mathbf{x}'} - q_{ideal,\mathbf{x}'}|$ is not negligible. This concludes the proof of the lemma. ∎

We remark that the previous lemma indicates that G-Independence is not only weaker than the other definitions, but also rather unsatisfactory. Indeed, by following G-Independence, we may deem protocols like Π_G "secure", when in reality they fail to provide even a very intuitive notion of independence – namely the one that requires the announced bits *do not always* sum 0. We stress the above result holds even for the uniform distribution.

6.2 Feasibility of CR and G independence

At this point, we have all the tools needed to prove the feasibility results for CR and G-Independence, namely that there exist protocols that achieve $(\mathcal{D}(\mathsf{CR}), \mathsf{CR})$-Independence as well as $(\mathcal{D}(\mathsf{G}), \mathsf{G})$-Independence. Indeed, Corollary 5.5 together with Claim 5.6 and the results of this section provide concise proofs for Claim 5.1 and Claim 5.3. Claim 5.1 follows from the existence of a protocol achieving $(\mathcal{D}(\mathsf{Sb}), \mathsf{Sb})$-Independence (by Corollary 5.5), and that $(\mathcal{D}(\mathsf{CR}), \mathsf{Sb})$-Independence implies

$(\mathcal{D}(\mathsf{CR}), \mathsf{CR})$-Independence. Claim 5.3 is proved analogously.

7 Other Issues and Open Problems

Number of Parties polynomially related to security parameter: Our results have been presented in the setting where there is constant number of parties n. Extensions of these results to the case n polynomially related to the security parameter are possible, but they require substantial changes to the definitions of [CR87, Gen00] (which assume the number of parties is fixed) and therefore, were out of the scope of this work.

Efficient Sb-Independent protocols: An interesting open problem is to find a constant round protocol (i.e., as efficient as the one of [Gen00]) for simultaneous broadcast that achieves the stronger notion of CR-Independence [CR87] or even (and preferably) Sb-Independence [CGMA85].

8 Acknowledgments

We thank Rosario Gennaro for helping us understand his work. We also would like to thank several anonymous referees for their help in improving the quality of this paper.

References

[BGW88] M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for noncryptographic fault-tolerant distributed computations. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing*, pages 1–10. ACM Press, 1988.

[BOEY03] M. Ben-Or and R. El-Yaniv. Resilient-optimal interactive consistency in constant time. *Distributed Computing*, 16(4), 2003.

[Can00] R. Canetti. Security and composition of multiparty cryptographic protocols. *Journal of Cryptology*, 13(1):143–202, 2000.

[CCD88] D. Chaum, C. Crepeau, and I. Damgård. Multiparty unconditional secure protocols. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing*, pages 11–19. ACM Press, 1988.

[CGMA85] B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch. Verifiable secret sharing and achieving simultaneity in the presence of faults. In *Proceedings of the 26th Annual Symposium on Foundations of Computer Science*, pages 383–395. IEEE Computer Society Press, 1985.

[CKPS01] C. Cachin, K. Kursawe, F. Petzold, and V. Shoup. Secure and efficient asynchronous broadcast protocols. In *Advances in Cryptology – CRYPTO ' 2001*, volume 2139 of *LNCS*, pages 524–541. Springer-Verlag, 2001.

[CR87] B. Chor and M. O. Rabin. Achieving independence in logarithmic number of rounds. In *Proceedings of the 6th Annual ACM Symposium on Principles of Distributed Computing*, pages 260–268. ACM Press, 1987.

[CR93] R. Canetti and T. Rabin. Fast asynchronous Byzantine agreement with optimal resilience (extended abstract). In *Proceedings of the 25th Annual ACM Symposium on the Theory of Computing*, pages 42–51, 1993.

[DDN01] D. Dolev, C. Dwork, and M. Naor. Nonmalleable cryptography. *SIAM Journal on Computing*, 30(2):391–437, April 2001.

[FM85] P. Feldman and S. Micali. Byzantine agreement in constant expected time (and trusting no one). In *Proceedings of the 26th Annual Symposium on Foundations of Computer Science*, pages 267–276. IEEE Computer Society Press, 1985.

[Gen95] R. Gennaro. Achieving independence efficiently and securely. In *Proceedings of the 14th Annual ACM symposium on Principles of Distributed Computing*, pages 130–136. ACM Press, 1995.

[Gen00] R. Gennaro. A protocol to achieve independence in constant rounds. *IEEE Transactions on Parallel and Distributed Systems*, 11(7):636–647, July 2000.

[GMW87] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game — A completeness theorem for protocols with honest majority. In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing*, pages 218–229. ACM Press, 1987.

[Gol01] Oded Goldreich. *Foundations of Cryptography: Basic Tools (Appendix)*. Cambridge University Press, 2001.

[LLM+01] M. Liskov, A. Lysyanskaya, S. Micali, L. Reyzin, and A. Smith. Mutually independent commitments. In *Advances in Cryptology – ASIACRYPT'01*, LNCS. Springer-Verlag, 2001.

[LSP82] L. Lamport, R. Shostak, and M. Pease. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, July 1982.

[PSL80] M. Pease, R. Shostak, and L. Lamport. Reaching agreements in the presence of faults. *Journal of the ACM*, 27(2):228–234, April 1980.

[Yao86] A. C.-C. Yao. How to generate and exchange secrets. In *Proceedings of the 27th Annual Symposium on Foundations of Computer Science*, pages 162–167. IEEE Computer Society Press, 1986.

A Alternative Characterization of Notions

A.1 Sb-Independence

We prove here that Sb-Independence is equivalent to (All, Sb)-Independence.

Proposition A.1 A protocol Π achieves Sb-Independence if and only if Π achieves (All, Sb)-Independence. ∎

Proof: One direction is trivial since Singleton $\subset$ All. We prove the other direction, namely that (Singleton, Sb)-Independence implies (All, Sb)-Independence. Assume Π is Sb-Independent. Then, for every polynomial-time adversary A attacking Π there exists a simulator S for $\mathsf{Ideal}(f_{SB})$. Then, by conditioning on the success probability of the distinguisher T on each particular value of the input distribution we get

$$\begin{aligned}
&\Big|\Pr\Big[\mathbf{x} \xleftarrow{R} \mathcal{D}^{(k)} \,:\, T(1^k, z, \mathbf{x}, \textsc{Exec}_A^{\Pi}(k, z, \mathbf{x})) = 1\Big] \\
&\qquad - \Pr\Big[\mathbf{x} \xleftarrow{R} \mathcal{D}^{(k)} \,:\, T(1^k, z, \mathbf{x}, \textsc{Exec}_S^{\mathsf{Ideal}(f_{SB})}(k, z, \mathbf{x})) = 1\Big]\Big| \\
&= \sum_{\mathbf{y}\in\{0,1\}^n} \Big(\Pr\Big[T(1^k, z, \mathbf{x}, \textsc{Exec}_A^{\Pi}(k, z, \mathbf{y})) = 1\Big] - \Pr\Big[T(1^k, z, \mathbf{x}, \textsc{Exec}_S^{\mathsf{Ideal}(f_{SB})}(k, z, \mathbf{y})) = 1\Big]\Big) \\
&\qquad \cdot \Pr\Big[\mathcal{D}^{(k)} = \mathbf{y}\Big] \;\leq\; \sum_{\mathbf{y}\in\{0,1\}^n} \big(k^{-c}\big) \cdot \Pr\Big[\mathcal{D}^{(k)} = \mathbf{y}\Big] \;=\; k^{-c}\,.
\end{aligned}$$

for some constant $c > 0$ and infinitely many values of k. Notice that the last inequality follows from the Sb-Independence of Π. This proves the result. ∎

A.2 CR-Independence

In this section, we present the definition of independence of [CR87], slightly generalized for arbitrary input distributions.

Definition A.2 [CR87] Let $\mathcal{D}$ be an input distribution over $\{0,1\}^n$. A protocol Π achieves independence under input distribution $\mathcal{D}$ if, for all adversary A, all honest party P_i, all "good" polynomial-time predicate R, all constant $c > 0$, and all sufficiently large k,

$$\left|\Pr\left[\,\mathbf{W}_i = 0\,\right] - \Pr\left[\,W_i = 0 \mid\ R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right]\right| \quad < \quad k^{-c} \tag{8}$$

where $\mathbf{W} \leftarrow \textsc{Announced}_A^{\Pi}(\mathcal{D})$. Predicate R is "good" if it occurs with non-negligible probability under adversary A and distribution $\mathcal{D}$. Formally, for $i \in [n]$ the polynomial-time computable predicate $R(W_1, \ldots,\ W_{i-1}, W_{i+1}, \ldots, W_n)$ is said to be *good* with respect to adversary A and distribution $\mathcal{D}$ if whenever party P_i is honest the event $R(\mathbf{W}_{\overline{\{i\}}}) = 1$ happens with non-negligible probability. That is,

$$\Pr\left[\,\mathbf{W} \leftarrow \textsc{Announced}_A^{\Pi}(\mathcal{D}^{(k)})\ :\ R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right]$$

is non-negligible in the security parameter k. ▌

We now show that Definition 4.3 and Definition A.2. are in fact equivalent.

Proposition A.3 A protocol Π achieves CR-Independence if and only if Π achieves independence under Definition A.2. ▌

Proof: We first prove that Definition 4.3 implies Definition A.2. Let Π be a parallel broadcast that achieves CR-Independence. It follows that, in particular, for all "good" predicates R

$$\left|\Pr\left[\,\mathbf{W}_i = 0\,\right] \cdot \Pr\left[\,R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right] - \Pr\left[\,W_i = 0 \ \wedge\ R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right]\right|$$

is negligible. Using this and that R is good, we get that the quantity

$$\begin{aligned}&\left|\Pr\left[\,\mathbf{W}_i = 0\,\right] - \Pr\left[\,\mathbf{W}_i = 0 \mid\ R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right]\right| \\ &\quad = \quad \Pr\left[\,R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right]^{-1} \cdot \left|\Pr\left[\,\mathbf{W}_i = 0\,\right] \cdot \Pr\left[\,R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right] - \Pr\left[\,\mathbf{W}_i = 0 \ \wedge\ R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right]\right|\end{aligned}$$

is negligible too.

We now prove that Definition A.2 implies Definition 4.3. Indeed, assume Π be a parallel broadcast that achieves independence according to Definition A.2. We analyze four cases depending on the probability p that the event $R(\mathbf{W}_{\overline{\{i\}}}) = 1$ (when $\mathbf{W} \leftarrow \textsc{Announced}_A^{\Pi}(\mathcal{D}^{(k)})$) occurs.

Case (a): p is negligible but non-zero, Then,

$$\begin{aligned}&\left|\Pr\left[\,\mathbf{W}_i = 0\,\right] \cdot \Pr\left[\,R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right] - \Pr\left[\,\mathbf{W}_i = 0 \ \wedge\ R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right]\right| \\ &\quad = \quad \Pr\left[\,R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right] \cdot \left|\Pr\left[\,\mathbf{W}_i = 0\,\right] - \Pr\left[\,\mathbf{W}_i = 0 \mid\ R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right]\right|\end{aligned} \tag{9}$$

is negligible, since $p = \Pr\left[\,R(\mathbf{W}_{\overline{\{i\}}}) = 1\,\right]$ is so.

Case (b): p is non-negligible. Since Π satisfies Definition A.2, the rightmost factor of Equation 9 is negligible too.

Case (c): $p = 0$, that is, R does never happen. Then, $\Pr\left[\, \mathbf{W}_i = 0 \,\wedge\, R(\mathbf{W}_{\overline{\{i\}}}) = 1 \,\right] = 0$, and

$$\left|\Pr\left[\, \mathbf{W}_i = 0 \,\right] \cdot \Pr\left[\, R(\mathbf{W}_{\overline{\{i\}}}) = 1 \,\right] - \Pr\left[\, \mathbf{W}_i = 0 \,\wedge\, R(\mathbf{W}_{\overline{\{i\}}}) = 1 \,\right]\right| \;=\; 0\,.$$

Case (d): p is neither negligible nor non-negligible. We prove the contrapositive. If Definition 4.3 does not hold for such R, there exist a constant $c > 0$ such that for infinitely many values of k

$$\left|\Pr\left[\, \mathbf{W}_i = 0 \,\right] \cdot \Pr\left[\, R(\mathbf{W}_{\overline{\{i\}}}) = 1 \,\right] - \Pr\left[\, \mathbf{W}_i = 0 \,\wedge\, R(\mathbf{W}_{\overline{\{i\}}}) = 1 \,\right]\right| \;\geq\; k^{-c} \tag{10}$$

Let S be the set of all values of k for which Equation (10) holds. Now, consider the following relation R' which equals R when $k \in S$ and equals one (ie. it is true) otherwise. Then, clearly R' is non-negligible and

$$\begin{aligned}
&\left|\Pr\left[\, \mathbf{W}_i = 0 \,\right] - \Pr\left[\, \mathbf{W}_i = 0 \mid\, R'(\mathbf{W}_{\overline{\{i\}}}) = 1 \,\right]\right| \\
&\quad= \Pr\left[\, R'(\mathbf{W}_{\overline{\{i\}}}) = 1 \,\right]^{-1} \cdot \left|\Pr\left[\, \mathbf{W}_i = 0 \,\right] \cdot \Pr\left[\, R'(\mathbf{W}_{\overline{\{i\}}}) = 1 \,\right] - \Pr\left[\, \mathbf{W}_i = 0 \,\wedge\, R(\mathbf{W}_{\overline{\{i\}}}) = 1 \,\right]\right| \\
&\quad\geq \left|\Pr\left[\, \mathbf{W}_i = 0 \,\right] \cdot \Pr\left[\, R(\mathbf{W}_{\overline{\{i\}}}) = 1 \,\right] - \Pr\left[\, \mathbf{W}_i = 0 \,\wedge\, R(\mathbf{W}_{\overline{\{i\}}}) = 1 \,\right]\right| \\
&\quad\geq k^{-c}
\end{aligned}$$

for all (infinitely many) $k \in S$. The result then holds. ∎

A.3 G-Independence

In this section, we present two equivalent notions of independence, and then show they imply G-Independence. Our first definition is expressed in terms of distributions ensembles.

Definition A.4 (G*-Independence) Protocol Π achieves G*-independence if for all adversaries A corrupting parties in $B \subset [n]$ (where $|B| = t < n$), for each corrupted party P_i, the ensembles (indexed by $k \in \mathbf{N}$, $\mathbf{x} \in \{0,1\}^n$, and $z \in \{0,1\}^*$)

$$\begin{aligned}
E &\stackrel{\text{def}}{=} \left\{ \mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{x}) \;:\; \mathbf{W_i} \right\} \\
E_0 &\stackrel{\text{def}}{=} \left\{ \mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{x}_B \sqcup \langle \mathbf{0} \rangle_{\overline{B}}) \;:\; \mathbf{W_i} \right\}
\end{aligned}$$

are statistically close (in the security parameter k). ∎

Our second definition, although more technical, is useful when proving implications or separations between G and other notions.

Definition A.5 (G**-Independence) Protocol Π achieves G**-independence if for all adversaries A corrupting parties in $B \subset [n]$ (where $|B| = t < n$), for each corrupted party P_i, for all vectors $\mathbf{r}, \mathbf{s} \in \{0,1\}^{\overline{B}}$, all vectors $\mathbf{w} \in \{0,1\}^B$, and all auxiliary input $z \in \{0,1\}^*$, the quantity

$$\left|\Pr\left[\, \mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{w} \sqcup \mathbf{s}) :\; \mathbf{W_i} = 1 \,\right] - \Pr\left[\, \mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{w} \sqcup \mathbf{r}) :\; \mathbf{W_i} = 1 \,\right]\right|$$

is negligible in the security parameter k. ∎

The two definitions are equivalent.

Proposition A.6 Let Π be a correct parallel broadcast. Then, Π achieves G^{**}-Independence if and only if Π achieves G^{*}-Independence. ▮

Proof:

$\mathsf{G}^{**} \Rightarrow \mathsf{G}^{*}$: Assume Π is not G^{*}-Independent. We want to prove Π is not G^{**}-Independent. Indeed, if Π is not G^{*}-Independent then ensembles E and E_0 must not be statistically close, and there exists $\mathbf{x} \in \{0,1\}^n$, $k \in \mathbf{N}$, and $z \in \{0,1\}^*$, for which there exists a constant $c > 0$ and infinitely many k such that (w.l.o.g.)

$$\Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{x}_B \sqcup \mathbf{x}_{\overline{B}}) \,:\, \mathbf{W_i} = 1\right] - \Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{x}_B \sqcup \mathbf{0}_{\overline{B}}) \,:\, \mathbf{W_i} = 1\right] \;>\; k^{-c}$$

The result follows immediately from taking $\mathbf{w} = \mathbf{x}_B$, $\mathbf{r} = \mathbf{x}_{\overline{B}}$ and $\mathbf{s} = \mathbf{0}_{\overline{B}}$.

$\mathsf{G}^{*} \Rightarrow \mathsf{G}^{**}$: Assume Π is not G^{**} independent. We want to prove Π is not G^{*}-Independent. Indeed, if Π is not G^{**}-Independent then there exists a vector $\mathbf{w} \in \{0,1\}^B$, distinct vectors $\mathbf{r}, \mathbf{s} \in \{0,1\}^{\overline{B}}$, an integer $k \in \mathbf{N}$, and a string $z \in \{0,1\}^*$, for which there exists a constant $c > 0$ and infinitely many k such that (w.l.o.g.)

$$\left|\Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{w} \sqcup \mathbf{s}) \,:\, \mathbf{W_i} = 1\right] - \Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{w} \sqcup \mathbf{r}) \,:\, \mathbf{W_i} = 1\right]\right| \;>\; k^{-c} \tag{11}$$

For simplicity, we define $D(\mathbf{a}) \stackrel{\text{def}}{=} \Pr\left[\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{a}) \,:\, \mathbf{W_i} = 1\right]$, for any vector $\mathbf{a} \in \{0,1\}^n$. Let $\mathbf{x} \stackrel{\text{def}}{=} \mathbf{w} \sqcup \mathbf{r}$ and $\mathbf{x}' \stackrel{\text{def}}{=} \mathbf{w} \sqcup \mathbf{s}$. Then, Equation (11) can be rewritten as $|D(\mathbf{x}) - D(\mathbf{x}')| > k^{-c}$. In consequence, using that $\mathbf{x}_B = \mathbf{x}'_B = \mathbf{w}$ we have

$$\left|D(\mathbf{x}) - D(\mathbf{x}_B \sqcup \mathbf{0}_{\overline{B}})\right| + \left|D(\mathbf{x}') - D(\mathbf{x}'_B \sqcup \mathbf{0}_{\overline{B}})\right| \;\;\geq\;\; \left|D(\mathbf{x}) - D(\mathbf{x}')\right| \;>\; k^{-c}$$

which implies that either $|D(\mathbf{x}) - D(\mathbf{x}_B \sqcup \mathbf{0}_{\overline{B}})| > k^{-c}/2$ or $|D(\mathbf{x}') - D(\mathbf{x}'_B \sqcup \mathbf{0}_{\overline{B}})| > k^{-c}/2$, and the result follows. ▮

The following result shows that both G^{*} and G^{**}-Independence imply G-Independence for any distribution for which G can be achieved.

Proposition A.7 , If a protocol Π achieves G^{**}-Independence then Π achieves G-Independence for any distribution $\mathcal{D} \in \Psi_{L,n}$. ▮

Proof: Let $\mathcal{D}$ be an arbitrary distribution in $\Psi_{C,n}$ and $\mathbf{r}, \mathbf{s} \in \{0,1\}^{n-t}$ two strings such that the probability $\mathcal{D}_{\overline{B}}$ equals $\mathbf{r}$ or $\mathbf{s}$ is not null. Also, let A be an arbitrary polynomial-time adversary that corrupt players in B ($t = |B|$) and let $i \in B$. For fixed values of $k \in \mathbf{N}$ and $z \in \{0,1\}^*$, we denote by $\Pr_{\mathcal{D},A}[E]$ the probability of event E under the choice $\mathbf{x} \stackrel{R}{\leftarrow} \mathcal{D}^{(k)}$ and $\mathbf{W} \leftarrow \textsc{Announced}^{\Pi}_{A(k,z)}(\mathbf{x})$. Now, for simplicity, we define the quantities

$$P(\mathbf{a},\mathbf{b}) \;\stackrel{\text{def}}{=}\; \Pr_{\mathcal{D},A}\left[\mathbf{W_i} = 1 \mid \mathbf{x}_B = \mathbf{a} \wedge \mathbf{x}_{\overline{B}} = \mathbf{b}\right]$$

$$Q(\mathbf{a},\mathbf{b}) \;\stackrel{\text{def}}{=}\; \Pr\left[\mathbf{x} \stackrel{R}{\leftarrow} \mathcal{D}^{(k)} \,:\, \mathbf{x}_B = \mathbf{w} \mid \mathbf{x}_{\overline{B}} = \mathbf{b}\right].$$

First, notice that $\mathbf{W}_{\overline{B}} = \mathbf{x}_{\overline{B}}$, since all uncorrupted parties always output their inputs. Then

$$\begin{aligned} &\Pr_{\mathcal{D},A}\left[\,\mathbf{W_i} = 1 \mid \mathbf{W}_{\overline{B}} = \mathbf{r}\,\right] - \Pr_{\mathcal{D},A}\left[\,\mathbf{W_i} = 1 \mid \mathbf{W}_{\overline{B}} = \mathbf{s}\,\right] \\ &= \sum_{\mathbf{w}\in\{0,1\}^t} \left(P(\mathbf{w},\mathbf{r})\cdot Q(\mathbf{w},\mathbf{r}) - P(\mathbf{w},\mathbf{s})\cdot Q(\mathbf{w},\mathbf{s})\right) \end{aligned} \tag{12}$$

By G^{**}-Independence, for all $\mathbf{w}' \in \{0,1\}^t$, all $\mathbf{r}', \mathbf{s}' \in \{0,1\}^{n-t}$, $|P(\mathbf{w}',\mathbf{r}') - P(\mathbf{w}',\mathbf{s}')| < \epsilon(k)$ where $\epsilon(k)$ is some negligible function in k. Let $P(\mathbf{w},\mathbf{t}^*) \stackrel{\text{def}}{=} \max_{\mathbf{t}^*}\{P(\mathbf{w},\mathbf{t})\}$. Then, by definition it follows that $P(\mathbf{w},\mathbf{r}) \leq P(\mathbf{w}^*,\mathbf{t})$ and, by G^{**}-Independence, that $P(\mathbf{w},\mathbf{s}) < P(\mathbf{w},\mathbf{t}^*) + \epsilon(k)$. Then, plugging these in Equation (12) we have

$$\sum_{\mathbf{w}\in\{0,1\}^t} \left(P(\mathbf{w},\mathbf{r})\cdot Q(\mathbf{w},\mathbf{r}) - P(\mathbf{w},\mathbf{s})\cdot Q(\mathbf{w},\mathbf{s})\right) \leq \sum_{\mathbf{w}\in\{0,1\}^t} \left(P(\mathbf{w},\mathbf{t}^*)\cdot\left(Q(\mathbf{w},\mathbf{r}) - Q(\mathbf{w},\mathbf{r})\right)\right) + \epsilon(k) \tag{13}$$

Now, define $R(\mathbf{w},\mathbf{r},\mathbf{s}) \stackrel{\text{def}}{=} Q(\mathbf{w},\mathbf{r}) - Q(\mathbf{w},\mathbf{s})$. We claim that $|R(\mathbf{w},\mathbf{r},\mathbf{s})|$ is negligible in k. Indeed, since $\mathcal{D} \in \Psi_{L,n}$,

$$\begin{aligned} |R(\mathbf{w},\mathbf{r},\mathbf{s})| &= |Q(\mathbf{w},\mathbf{r}) - Q(\mathbf{w},\mathbf{s})| = \left|Q(\mathbf{w},\mathbf{r}) - \Pr\left[\,\mathcal{D}_B^{(k)} = \mathbf{w}\,\right] + \Pr\left[\,\mathcal{D}_B^{(k)} = \mathbf{w}\,\right] - Q(\mathbf{w},\mathbf{s})\right| \\ &\leq \left|\Pr\left[\,\mathbf{x} \stackrel{R}{\leftarrow} \mathcal{D}^{(k)} \,:\, \mathbf{x}_B = \mathbf{w} \mid \mathbf{x}_{\overline{B}} = \mathbf{r}\,\right] - \Pr\left[\,\mathcal{D}_B^{(k)} = \mathbf{w}\,\right]\right| \\ &\quad + \left|\Pr\left[\,\mathbf{x} \stackrel{R}{\leftarrow} \mathcal{D}^{(k)} \,:\, \mathbf{x}_B = \mathbf{w} \mid \mathbf{x}_{\overline{B}} = \mathbf{r}\,\right] - \Pr\left[\,\mathcal{D}_B^{(k)} = \mathbf{w}\,\right]\right| \\ &< 2\cdot\epsilon'(k) \end{aligned} \tag{14}$$

for some negligible function $\epsilon'(k)$. Combining Equations (12), (13) and (14) we obtain

$$\left|\Pr\left[\,\mathbf{W_i} = 1 \mid \mathbf{W}_{\overline{B}} = \mathbf{r}\,\right] - \Pr\left[\,\mathbf{W_i} = 1 \mid \mathbf{W}_{\overline{B}} = \mathbf{s}\,\right]\right| < 2\cdot\epsilon'(k) + \epsilon(k)\,.$$

This proves the result. ∎

B Secure Function Evaluation

We briefly recall the definition of secure function evaluation from [Can00]. Let $n \in \mathbf{N}$ be a fixed parameter. Assume Π is an n-party protocol, $\mathbf{x} = (x_1,\ldots,x_n)$ is a vector of inputs for the parties, and $f\colon (\{0,1\}^*)^n \to (\{0,1\}^*)^n$ be a function that map n strings into n strings. Intuitively, the goal of protocol Π is to compute function f on the inputs held by the parties, in such a way that each party receives some output as specified by f. In order to define the security of a protocol computing f, the simulation paradigm is used and two processes or "worlds" are considered: an ideal process and a real process.

In the ideal process, there is a trusted third party connected to each party by a point-to-point channel. The trusted party may compute f on the inputs provided by the parties and returns the output to the parties as specified by f. In this world, the protocol $\mathsf{Ideal}(f)$ that computes f on the parties' input vector $\mathbf{x}$ is very simple: all parties privately submit their inputs to the trusted party, which computes $(y_1,\ldots,y_n) = f(\mathbf{x})$ and returns each output y_i to party P_i. Adversaries in the ideal process cannot corrupt the trusted party but can corrupt an arbitrary subset B of the parties before the protocol starts. Upon corruption, party P_j gives her input x_j to the adversary and

all her outgoing messages and output is controlled by the adversary. The communication channel between each honest party and the trusted third party cannot be eavesdropped by the adversary. This process models the ideal case in which an adversary cannot disrupt the computation other than replacing the inputs and outputs of corrupted parties, nor obtain more information from the inputs and outputs of the honest parties other than what can be inferred from the output of the corrupt parties.

In the real process there is no trusted third party and parties are connected by pairwise point-to-point channels. Adversaries in the real process can also corrupt an arbitrary set B of parties; corruption occurs as in the ideal process. Parties execute protocol Π in this world.

For any security parameter $k \in \mathbf{N}$, any input $\mathbf{x} = (x_1, \ldots, x_n) \in (\{0,1\}^*)^n$ for the parties, and any real-process adversary A with auxiliary input $z \in \{0,1\}^*$, we let $\text{EXEC}_A^\Pi(k, \mathbf{x})$ denote the $(n+1)$-vector whose elements are the output of each party and the adversary after executing protocol Π under adversary A. That is,

$$\text{EXEC}_A^\Pi(k, z, \mathbf{x}) \stackrel{\text{def}}{=} \left(\text{OUTPUT}_A^\Pi(k, z), \text{OUTPUT}_{P_1}^\Pi(k, x_1), \ldots, \text{OUTPUT}_{P_n}^\Pi(k, x_n) \right) .$$

where $\text{OUTPUT}_C^\Sigma(k, x)$ denotes the local output of entity C (either a party or adversary) after executing protocol Σ on input the security parameter k and value x. Similarly, for the ideal process, given any ideal-process adversary S the execution of $\mathsf{Ideal}(f)$ under adversary S is defined as

$$\text{EXEC}_S^{\mathsf{Ideal}(f)}(k, z, \mathbf{x}) \stackrel{\text{def}}{=} \left(\text{OUTPUT}_S^{\mathsf{Ideal}(f)}(k, z), \text{OUTPUT}_{P_1}^{\mathsf{Ideal}(f)}(k, x_1), \ldots, \text{OUTPUT}_{P_n}^{\mathsf{Ideal}(f)}(k, x_n) \right) .$$

Both quantities define ensembles in the straightforward way,

$$\begin{aligned} \text{EXEC}_A^\Pi &\stackrel{\text{def}}{=} \left\{ \text{EXEC}_A^\Pi(k, z, \mathbf{x}) \right\}_{k \in \mathbf{N}, z \in \{0,1\}^*, \mathbf{x} \in (\{0,1\}^*)^n} \\ \text{EXEC}_S^{\mathsf{Ideal}(f)} &\stackrel{\text{def}}{=} \left\{ \text{EXEC}_S^{\mathsf{Ideal}(f)}(k, z, \mathbf{x}) \right\}_{k \in \mathbf{N}, z \in \{0,1\}^*, \mathbf{x} \in (\{0,1\}^*)^n} \end{aligned}$$

A protocol Π securely implements function f if for any real-process PPT adversary A there exists an ideal-process PPT adversary S such that EXEC_A^Π is computationally close to $\text{EXEC}_S^{\mathsf{Ideal}(f)}$, denoted

$$\text{EXEC}_A^\Pi \stackrel{c}{\approx} \text{EXEC}_S^{\mathsf{Ideal}(f)}$$

Spelled out, protocol Π securely implements function f if for any real-process PPT adversary A there exists an ideal-process PPT S such that for all PPT distinguishers D, for all constant $c > 0$ and for all sufficiently large k, all $z \in \{0,1\}^*$ and $\mathbf{x} \in (\{0,1\}^*)^n$,

$$\left| \Pr \left[D(1^k, z, \mathbf{x}, \text{EXEC}_A^\Pi(k, z, \mathbf{x})) = 1 \right] - \Pr \left[D(1^k, z, \mathbf{x}, \text{EXEC}_S^{\mathsf{Ideal}(f)}(k, z, \mathbf{x})) = 1 \right] \right| \quad < \quad k^{-c}$$

C Some useful relations

Let $\mathbf{X} = (X_1, \ldots, X_n)$ an arbitrary random variable that takes values in $\{0,1\}^n$. For any set $G \subset \{1, \ldots, n\}$, any $i \notin G$, any bit b, and any string $u \in \{0,1\}^{n-t}$ such that $\Pr[\mathbf{X}_G = u] > 0$, let $p_{b,i}$ and $q_{b,i}u$ denote the values $p_{b,i} \stackrel{\text{def}}{=} \Pr[X_i = b]$ and $q_{b,i}(u) \stackrel{\text{def}}{=} \Pr[X_i = b \mid \mathbf{X}_G = u]$. Furthermore, let $\epsilon(k)$ be a negligible function, that is, $\epsilon(k) < k^{-c}$ for all constant $c > 0$ and sufficiently large k.

Proposition C.1 Assume, $|p_{b,i} - q_{b,i}(u)| < \epsilon(k)$ for all $u \in \{0,1\}^{n-t}$. Then $|q_{b,i}(r) - q_{b,i}(s)| < 2 \cdot \epsilon(k)$ for all $r, s \in \{0,1\}^{n-t}$. ∎

Proof: There are four cases, depending on whether $p_{b,i}$ is larger or smaller than $q_{b,i}(r)$ and $q_{b,i}(s)$. Assume $p_{b,i} < q_{b,i}(r)$ and $p_{b,i} < q_{b,i}(s)$. Then, $q_{b,i}(r) - p_{b,i} < \epsilon(k)$ and $q_{b,i}(s) - p_{b,i} > 0$. Subtracting and taking absolute value we get $|q_{b,i}(r) - q_{b,i}(s)| < \epsilon(k)$ and the result holds. The case $p_{b,i} > q_{b,i}(r)$ and $p_{b,i} > q_{b,i}(s)$ is analogous.

Now, assume $q_{b,i}(r) > p_{b,i}$ but $p_{b,i} > q_{b,i}(s)$. Then, $q_{b,i}(r) - p_{b,i} < \epsilon(k)$ and $p_{b,i} - q_{b,i}(s) < \epsilon(k)$. Subtracting and taking absolute value we get the result. The case $p_{b,i} > q_{b,i}(r)$ and $p_{b,i} < q_{b,i}(s)$ is analogous. ▌