REALIZING SECURE MULTIPARTY COMPUTATION ON INCOMPLETE NETWORKS

Shailesh Vaya
Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai 600036, India

Keywords: Incomplete networks, Byzantine adversary, Almost everywhere secure computation, Hybrid argument, Input indistinguishability.

Abstract: Secure multiparty computation of a multivariate function is a central problem in cryptography. It is well known that secure multiparty computation can be realized by a set of n parties iff the connectivity of the underlying (authenticated) communication network is more than twice the number of corrupted parties. This impossibility result makes secure multiparty computation far less applicable in practice, as most deployed networks have a degree much lower than $O(n)$ and one would ideally like to be able to tolerate $\Theta(n)$ corrupted parties. In this work we consider a recently proposed model for (Unconditional) secure multiparty computation for networks of low degrees for which authenticated channels are available only between very few pairs of parties. Not all honest parties may be expected to achieve traditional security guarantees of multiparty computation for this model because of theoretical limitations posed by lack of infrastructure. Honest parties which are not given canonical guarantees of Correctness and Privacy are called "sacrificed" and the resulting notion is called almost everywhere secure computation.
In this conceptual note, we investigate the previous results about this model by Garay and Ostrovsky'2008. We explain why these results hold only for weak honest-but-curious type passive corruptions. We emphasize why the results for almost everywhere secure computation are theoretically interesting and practically relevant only when active malicious corruptions are allowed. We argue the limitation of hybrid argument for realizing privacy property when Byzantine corruptions are allowed. From this we deduce that simulation based reduction approach is necessary to realize even input indistinguishability type definition of privacy for the stand alone setting. We present a conceptual exposition of the simulator based approach for defining privacy of a.e.s.c.. Finally, we present a brief overview of technicalities involved in realizing a.e.s.c. when malicious corruptions are allowed.

1 INTRODUCTION

In secure multiparty computation n players jointly evaluate an n-variate polynomial time computable function f on the vector of their input values. The strong guarantee for secure MPC is that even if a (small) fraction of the players are controlled by a malicious adversary, all the honest parties still obtain their correct output values. Furthermore, the computation process does not reveal any more information about the individual inputs of honest parties, than what can be inferred by the committed input values/output values of corrupted parties and the structure of the function f. Cryptographic solutions for MPC were first presented in (Yao, 1982) for the case of two parties, followed by (Goldreich et al., 1987) for n parties. (BenOr et al., 1988),(Chaum et al., 1988) presented solutions for information theoretic setting, assuming more than $\lfloor \frac{2*n}{3} + 1 \rfloor$ parties are honest. This work focuses on the information theoretic setting.

In (Vaya, 2007),(Garay and Ostrovsky, 2008) authors propose a relaxed notion of secure computation problem relevant to the setting of incomplete networks, called *almost everywhere secure computation* aka a.e.s.c. This notion is a generalization of almost everywhere agreement, proposed in (Dwork et al., 1988). Almost everywhere agreement is a type of Byzantine agreement in which some honest parties are not able to achieve the correct agreement value. This happens because of limitations on the topology of the underlying communication network. In incomplete networks of small degree for example, the communication of an honest party can be sabotaged by corrupting the small number of nodes in the neighborhood of the honest party. Such an honest party cannot correspond reliably with other honest parties and

Vaya S. (2010).
REALIZING SECURE MULTIPARTY COMPUTATION ON INCOMPLETE NETWORKS.
In *Proceedings of the International Conference on Security and Cryptography*, pages 270-277
DOI: 10.5220/0002986902700277

expect to achieve the agreement value. The resulting relaxed notion of agreement is called almost everywhere agreement. Due to identical reasons, some honest parties may not be able to achieve Correctness or Privacy (or both) Properties when protocols for multiparty computation are executed on the incomplete network. This relaxation of multiparty computation, relevant to the setting of incomplete networks, is called almost everywhere secure computation. Honest parties which are not guaranteed to achieve Correctness or Privacy (or both) for secure multiparty computation, are referred to as "sacrificed". Thus, a.e.s.c. says that as long as the fraction of maliciously corrupted parties t is small (i.e., smaller then a given value), a large number of (unsacrificed) honest parties (at least $\lfloor \frac{2*n}{3} \rfloor + 1$ in number) are assured of canonical guarantees of secure multiparty computation. For an elaborate motivational discussion on a.e.s.c., the reader is referred to (Garay and Ostrovsky, 2008).

(Garay and Ostrovsky, 2008) present Input indistinguishability type definition of privacy for a.e.s.c.. This approach to defining privacy for unconditional MPC, was originally proposed in (Kilian et al., 1994) for the vanilla model. A hybrid argument was presented to realize this definition for honest-but-curious type corruptions. GO08 adapt this approach to formulate definition of privacy for the framework of almost everywhere secure computation. They show how to realize a.e.s.c. on incomplete networks, when honest-but-curious type passive corruptions are allowed. This is a useful first step, however from a practical and theoretical perspective the study of a.e.s.c. (and for that matter secure multiparty computation) is interesting and relevant only when malicious corruptions (Byzantine corruptions in most general form) are allowed. When only mild passive corruptions are allowed, the important correctness property is trivially achieved by all parties, which sabotages a.e.s.c. in a fundamental way. Thus, the principle open problem in this line of research is whether a.e.s.c. can be meaningfully realized for general Byzantine corruptions. This is resolved in (Vaya, 2007).

1.1 Related Works

Realization of almost everywhere secure computation heavily uses the infrastructure developed in the seminal work on *almost everywhere agreement* by (Dwork et al., 1988). The notion of almost everywhere secure computation for incomplete networks was proposed in (Garay and Ostrovsky, 2008), (Vaya, 2007). Input indistinguishability type definition of privacy for a.e.s.c. was first proposed in (Kilian et al., 1994), in which results are given for honest-but-curious type of *passive corruptions* using a hybrid argument. Assuming more than $\lfloor \frac{2*n}{3} \rfloor$ parties are honest, it has been shown that it is possible to securely compute any n-variate function, (BenOr et al., 1988), (Chaum et al., 1988) in the information theoretic regime.

1.2 Outline of the Article

Following is a brief outline of this article. In Section 5, we elaborately investigate the results presented in (Garay and Ostrovsky, 2008). We discuss the relevance of the results on almost everywhere *private* computation presented in subsection 2. In Section 3, we rigorously analyze why a simplistic hybrid argument, employed in GO08, cannot be extended to realize the definition of privacy when malicious corruptions are allowed. This discussion emphasizes the necessity of a more sophisticated reduction argument to deal with the mathematically richer structure of views of parties/adversary, created when malicious corruptions are allowed. In Section 5, we present a short conceptual exposition of the simulation based definition of privacy of a.e.s.c. in GO08. Finally, in Section 5 we present a brief overview of the technicalities involved in correctly realizing a.e.s.c..

2 BRIEF DISCUSSION OF ALMOST EVERYWHERE PRIVATE COMPUTATION REALIZED IN (Garay and Ostrovsky, 2008)

The results in (Garay and Ostrovsky, 2008) have been discussed elaborately in (Vaya, 2010). This discussion emphasizes the importance of rigorously formulating and realizing almost everywhere secure computation for theoretically interesting and practically relevant case of malicious corruptions. Very briefly, the straightforward hybrid argument used in (Garay and Ostrovsky, 2008) cannot be extended to handle malicious corruptions. Simulation based reduction is necessary, to argue indistinguishability of distributions of mathematically richer structure of views of corrupted parties, when malicious corruptions are allowed, to realize any privacy definition (Pl. check Section 3 for relevant discussion).

For the sake of completeness, the proof of the main Theorem 4.3 in (Garay and Ostrovsky, 2008) has been presented and investigated in the Appendix.

3 NEED FOR SIMULATION BASED APPROACH FOR REALIZING PRIVACY

Definitions of privacy are based on indistinguishability of certain probability distributions. The discussion below has been presented keeping in mind the input indistinguishability type approach presented in (Kilian et al., 1994) for stand alone settings (and (Garay and Ostrovsky, 2008) for a.e.s.c.). We rigorously argue why straightforward hybrid argument is limited in dealing with complex mathematical structure of views of parties, generated from execution of unconditional MPC protocols when malicious corruptions are allowed.

Let C denote the subset of corrupted parties and $P-C$ be the subset of honest parties. Let the vector of initial input values and committed input values by the corrupted parties be $\overrightarrow{y}_C$ and $\overrightarrow{x}_C$ respectively and $o = f(\overrightarrow{x})$ be the output generated from the computation. Let $\mathbf{view}^{\mathcal{A}}_{\overrightarrow{y},\overrightarrow{x},o}$ denote the distribution of views of the adversary generated from executions of multiparty protocol, when parties start with initial input vector $\overrightarrow{y}$, commit to input vector $\overrightarrow{x}$ and compute output o. *Input indistinguishability* says that: As long as the adversary starts with the same initial input vector $\overrightarrow{y}_C$, commits to same input vector $\overrightarrow{x}_C$ and same output o is computed, the distribution of views of the adversary $\mathcal{A}$ generated is the same. This is true for all the different multiparty executions corresponding to different vectors of input values of honest parties $(\overrightarrow{y}^1_{P-C} = \overrightarrow{x}^1_{P-C}, \overrightarrow{y}^2_{P-C} = \overrightarrow{x}^2_{P-C}, \ldots)$. The underlying interpretation is that the view of the adversary is not anymore helpful in distinguishing between different candidates of input vectors of honest parties, than what can be inferred from vector of committed input value(s) of corrupted parties and the computed output value alone.

Now let us see why straightforward hybrid argument has limitations in realizing Input Indistinguishability type definition of privacy, when Byzantine corruptions are allowed. During the execution of (unconditional) secure multiparty protocols, parties commit to certain secrets (e.g., the input value during the commitment phase of BGW protocol). This is achieved by executing a protocol for secret sharing called Verifiable Secret Sharing (VSS): The committing party shares its secret input value by sending evaluations of a polynomial to the rest of the parties. The parties then exchange sub-shares of these shares between themselves to make sure that the secret value committed by the committing party is valid, and can be recovered by the honest parties. A malicious party may send incongruent shares of its secret/share-of-secret to other honest parties or other corrupted parties. When honest parties later tally the values of their sub-shares they would not concur. Alternatively, some corrupted parties may raise a false alarm about the mismatch of subshare values. The parties then seek to resolve the inconsistencies by executing a "disavowal" sub-protocol in which the committing party may be asked to (publicly) declare the correct shares/sub-shares of the contentious parties. The execution of this *disavowal* sub-protocol can happen due to variety of such reasons and results in execution of different/lengthier sequence of steps of the BGW protocol compared to when it is not executed at all. Whether the "disavowal" protocol is executed once or more, what leads to its execution, and how many times it is executed depends on the dynamic choices made by the corrupted parties during the execution of the protocol. Thus, the actual lengths and contents of the views of all parties, including the adversary, can vary drastically, from one execution of BGW protocol to another. This is true even if all the parameters (like initial value, committed value and output value) of the parties are same.

A *straightforward hybrid argument* cannot be applied to compare two probability distributions on variable length views of adversary. This is due to the following reason. A hybrid argument (HA) is used to argue indistinguishability of two random variables $X[1,\ldots,m]$ and $Y[1,\ldots,m]$ that have the same length/size m. A typical HA proceeds as follows: Let random variables $X[1,\ldots,m] \not\approx Y[1,\ldots,m]$. Then, for some $i \in [m] : X[i] \not\approx Y[i]$. It is now argued that $X[i] \not\approx Y[i]$ is not possible. From this it is concluded that $X[1,\ldots,m] \approx Y[1,\ldots,m]$. When only honest-but-curious type passive corruption is allowed the sizes of the views are fixed and such a HA can be employed to realize input indistinguishability. However, when malicious corruptions are allowed, the lengths of the views of the adversary can be variable. Thus, support of probability distributions on views of adversary from which X and Y are drawn, may consist of strings of vastly different lengths. Arguing indistinguishability of such random variables calls for a sequential inductive argument of the following type: First argue the indistinguishability of the first bit of two random variables. Then, conditioned on the fact that first bits are indistinguishable, argue the indistinguishability of the second two bits of the random variables. So on and so forth. In context of BGW protocol, the argument would proceed by comparing the distribution of views of adversary, generated from two different scenarios, after every 'round/super-round' of the protocol. We present such an argument in (Vaya, 2007) by adapting the concept of simulator inherent in defini-

tion of ZKP in a fundamentally new way.

The above argument has been presented catering to the vanilla model of unconditional MPC, (BenOr et al., 1988). However, exactly the same reasoning also holds for almost everywhere secure computation, which is only more complex then the vanilla MPC model. Thus, the proof of Theorem 4.3, (Garay and Ostrovsky, 2008), does not hold true when maliciously corrupted party raises a false alarm or sends incongruent shares of its secret during Verifiable secret sharing subprotocol.

4 EXPOSITION OF SIMULATOR BASED DEFINITION OF PRIVACY OF A.E.S.C.

Violating privacy property in the information theoretic setting amounts to an adversary inferring information about the input values of honest parties that it otherwise should not. Adversary can make any inference about the input values of other parties on the basis of its own view only. If it is shown that an indistinguishable distribution of views of the adversary can (always) be generated from a certain set of values, then it implies that the adversary can infer nothing more about the input values of unsacrificed honest parties than what can be information theoretically inferred on the basis of these values alone and the structure of the function f. Obviously, the adversary cannot distinguish between the different vectors of input values of unsacrificed honest, which are (equally) consistent with its own view, with any advantage. This is the principle on which the definition of privacy is based.

Let us review the requirement imposed by the definition of privacy in (Vaya, 2007): There exists a simulator *Sim*, such that for all corruptions $\overrightarrow{C} \in \mathcal{T}$, vectors of initial input values $\overrightarrow{y}$ and committed input values $\overrightarrow{x}$ the following holds true: $Sim^{\mathcal{A}}(\overrightarrow{C}, \mathcal{T}, N, \overrightarrow{y}_{P-H}, \overrightarrow{x}_{P-H}, f(\overrightarrow{x})) \approx \overrightarrow{View}_{\overrightarrow{C}}^{\Pi,\mathcal{A}}(\overrightarrow{C}, \overrightarrow{y}, \overrightarrow{x}, f(\overrightarrow{x}))$.

Thus, imagine a table of all plausible vectors of input values committed by all the participating parties (for unsacrificed honest parties the committed input values = initial input values, for the rest they may differ) and corresponding output values for the given multi-variate function f. The goal of the adversary is to zoom on to the smallest subset of table entries which are most-yet-equally consistent with its (distribution of) view(s) generated in real execution of the protocol. This set of compatible table entries correspond to the plausible vectors of input values that unsacrificed honest parties started with. Table entries which are incompatible (correspond to negligible probability of occurrence) with adversaries view are rejected. The guarantee of the definition of privacy, in (Vaya, 2007), is that this set of non-rejected table entries must certainly include the following subset of entries (as long as adversary is $\mathcal{T}$-restricted): *The subset of table entries which are compatible with the actual input values committed by the sacrificed honest parties, the corrupted parties and the output value generated in the real execution of the protocol.* Thus, fix the distribution of views of the adversary as $\approx Sim(\overrightarrow{I_i}, \overrightarrow{I_c}, Out)$ (where $\overrightarrow{I_i}$ is the vector of initial input values and $\overrightarrow{I_c}$ is the vector of committed input values of the corrupted and sacrificed parties) for some real execution of the multiparty protocol. Then, the adversary cannot distinguish between any two vectors $\overrightarrow{l}, \overrightarrow{m}$ of initial input values, with which unsacrificed honest parties could have started with, for which it is true that $f(\overrightarrow{l}, \overrightarrow{I_c}) = f(\overrightarrow{m}, \overrightarrow{I_c}) = Out$, with any useful advantage.

4.1 Remark by Canetti, Garay and Ostrovsky Regarding the Simulation based Approach

In (Garay and Ostrovsky, 2008), the authors make the following technical remark about the definition of privacy in (Vaya, 2007):

> It is well known that an information-theoretic definition of privacy in terms of indistinguishability is weaker than a simulation based counterpart. For example, consider a secure - according to our definition - multiparty protocol to compute $f(x)$ for a one way permutation f, where x should remain hidden from all the players. Information theoretically, the computation of $f(x)$ and the computation that reveals x reveals the same amount of information to an infinitely powerful adversary; however, in the latter case clearly x does not remain hidden. This example, due to Canetti, shows that one should not "mix" information theoretic notions with computational notions, and that only suitable properties, such as those guaranteed by information theoretically secure MPC protocols, will remain secure under our definition.

In the light of the preceding discussion it is easy to see that even if inputs of some of the "sacrificed" honest parties are not available to the adversary and hence the simulator *Sim*. However, the interpretation of

demonstrating such a simulator still holds true. More interestingly, we note that the semantic of "simulator" based definition implies the Input indistinguishability type definition of privacy presented in (Garay and Ostrovsky, 2008).

Why is Simulator sometimes Given Input Values, of some Sacrificed Honest Parties, which cannot be Extracted from Adversaries View? We give an example of a case, which arises in the analysis, that justifies why simulator may sometimes be given input values of some sacrificed honest parties, even though the value itself is not extractable from the adversaries view: The PSMT protocol can fail because of complex reasons. In particular, it can sometimes happen that the adversary is able to corrupt the message sent on the PSMT channel, while the message itself is information theoretically secured from the adversary. With sufficient number of such channels connected to an honest party (which in turn depends on the connectivity of the network), the Input commitment phase of the BGW/CCD protocol may fail so that the honest party is made to commit to the default value "d", which is learned by all the parties in the network. Thus, depending on the dynamic choices made by the adversary, the privacy of this honest party may be compromised sometimes, while is preserved at other times. However, since privacy property cannot be guaranteed for this honest party always, it is sacrificed. Thus, although simulator is given the input value of this (sacrificed) honest party always i.e., whenever the corresponding subset of parties are corrupted, the value itself is not always extractable from the view of the adversary.

5 REALIZING SECURE COMPUTATION ON INCOMPLETE NETWORKS WHEN BYZANTINE CORRUPTIONS ARE ALLOWED

The reader is referred to (Vaya, 2007), for complete details regarding realizing almost everywhere secure computation when Byzantine corruptions are allowed. In this section, we present a brief overview of the main technical subtleties involved.

The overall approach for realizing a.e.s.c. is as follows: (1) Construct a complete network N_C from the original incomplete network N by adding virtual channels between nodes that are not connected. (2) Adopt a standard proof of security of the BGW protocol to suit our definition of a.e.s.c.. (3) Order all the virtual channels in this network N_C. Now inductively (a) Replace each virtual channel in N_C with simulation of the corresponding PSMT protocol, on the original incomplete network (b) Prove security of the associated multiparty protocol for the intermediate network, using a reduction argument. For realizing this approach we encounter some technical challenges which influences the structure of definitions, protocols and the proofs, which are highlighted next.

The Definitional Approach and Tightness of Results. We first argue there is no clean and satisfactory way to formulate the definition of security using the Trusted third party paradigm, in which an Ideal world process is defined and compared to the Real world process: The honest parties, that are sacrificed, have a complicated interaction with the Ideal world functionality because some of the channels, realized using the infrastructure of the underlying incomplete network between distant parties, can be controlled to different extents (passive eavesdrop or active partial/full corruption) by the adversary. Depending on the dynamic influence of the adversary on these channels, some sacrificed honest parties may end up committing to different values which can be the default value d, some other value partially influenced or fully determined by the adversary or even the original value itself. Furthermore, depending on choices made by adversary this value may or may not be extractable from adversaries view. As also discussed in (Garay and Ostrovsky, 2008), we find that there is no clean and satisfactory way to formulate an Ideal world functionality that handles this complex interaction of the sacrificed honest parties with other parties which is also controlled by the adversary. In fact, depending on the topology, choice of protocol etc. scenarios may arise for which the final status of these parties with respect to sacrifice of privacy property may change. Furthermore, it is also not clear what output should be assigned by the Ideal functionality to the sacrificed parties, as it depends on how the adversary corrupts the shares of the output value sent to them.

Due to aforementioned reasons we find that we cannot invoke standard composition theorems in literature for this model. This is because composition theorems are statements about vector of distributions of views of honest and corrupted parties. However, here we may encounter sacrificed parties which do not fall in the category of either fully corrupted or fully honest parties. Without invoking these theorems the strongest results we can hope to achieve are for the stand alone setting. But without using these theorems, we have to compose the main protocol with several sub-protocols for PSMT for realizing virtual

channels, ourselves. This makes the proof involved even for the stand alone setting.

The Need for Adversary Structures and Virtual Channels. Observe that the behavior of the virtual channels is not fixed and may vary for the same topology of the underlying incomplete network. The behavior depends on the choice of subset of parties corrupted by the adversary and the actual PSMT protocol employed (Different PSMT protocols may utilize the infrastructure of the incomplete network differently). In particular, the same virtual channel may behave as a tamperable channel or as an eavesdroppable channel or as an uncorrupted authenticated channel. Even a small variation in the subset of parties corrupted by the adversary may influence a large number of virtual channels to behave differently. Thus, we are faced with exponential number of variations. Towards this end we set up adversary structures which fixes the behavior of the virtual channels for a given choice of subsets of parties corrupted by the adversary. Obviously for different topologies of the incomplete networks and different PSMT protocols the adversary structures are quite different. Adversary structures for which we cannot realize secure function evaluation are called infeasible.

The intermediate Networks. We replace one virtual channels at a time, from the original constructed network N_C, by simulation of PSMT protocol on the incomplete network. We prove that the resulting intermediate protocol for the intermediate network is secure. Our proof uses the reduction argument. For this purpose, we define the functionality of the virtual channels so that it provides the adversary with more flexibility, compared to the case when the virtual channel is substituted by execution of PSMT protocol on the incomplete network. For example, an adversary for the former case (of virtual channels) gets to see the message being sent on the virtual channels earlier, then it would in the latter case. Thus, we start by handling strictly more powerful adversaries while considering greater number of virtual channels and consider progressively weaker adversaries as we replace the virtual channels one by one. This is useful in reducing the complexity of the reduction argument.

Super-rounding and Slotting the Super-round. The original protocol is synchronous and proceeds in rounds. We adapt it to our setting of incomplete networks, where many pairs of nodes cannot communicate with each other in a single round. To take care of this, we expand each round of the original protocol to a super-round, so that distant parties can communicate with each other by executing PSMT protocol on the incomplete network. When replacing virtual channels of the complete network (as described above) by simulation of PSMT on the incomplete network, we encounter the following technical difficulty. *When transmissions take place on two virtual channels simultaneously, then the messages sent/received/heard on one channel may influence the corruption of messages on another channel and vice verse.* Handling this issue for the case of active corruptions makes the proof of security of the underlying protocol complex. We go around this problem at the expense of increase in the round complexity of the protocol by partitioning each super-round into slots so that each pair of parties is assigned a dedicated non-overlapping slot. The assigned time slots are wide enough to send messages on real edges or virtual channels. If there is a real edge in the underlying network for a given pair of parties then the message is sent in the first round of the slot. For virtual channels the allotted slot should be wide enough to execute any PSMT protocol on the underlying network.

Thus, we obtain $\mathcal{T}$-secure computation (technical name of a.e.s.c. in (Vaya, 2007)) on incomplete networks which possess a certain $\mathcal{T}$-Communicability property (which allows setting up of the requisite infrastructure on certain incomplete networks) against a $\mathcal{T}$-restricted adversary $\mathcal{A}$:

Theorem 5.1. *If network N possesses $\mathcal{T}$-Communicability Property, then there exists a two-phase multiparty protocol Π_N, that $\mathcal{T}$-securely evaluates function f, on network N.*

Full details of definitions and theorems are given in (Vaya, 2007).

ACKNOWLEDGEMENTS

I thank Dr. Pandurangan for a useful suggestion. I am grateful to Dr. Palash Sarkar and anonymous reviewers for their careful reading and invaluable comments, which have greatly improved the presentation of this article. I thank CS Department at IIT Madras for their invaluable support.

REFERENCES

Ben-Or, M., Goldwasser, S., and Wigderson, A. (1988). Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Symposium on Theory of Computation*, Chicago, Illinois. Association for Computing Machinery.

Chaum, D., Crepeau, C., and Damgard, I. (1988). Multiparty unconditionally secure protocols. In *Symposium*

on Theory of Computing, Chicago, Illinois. Association for Computing Machinery.

Dwork, C., Peleg, D., Pippinger, N., and Upfal, E. (1988). Fault tolerance in networks of bounded degree. *SIAM Journal on Computing*.

Garay, J. and Ostrovsky, R. (2008). Almost everywhere secure computation. In *Advances in Cryptology, EUROCRYPT*.

Goldreich, O., Micali, S., and Wigderson, A. (1987). How to play any mental game or a completeness theorem for protocols with honest majority. In *Symposium on Theory of Computation*. Association for Computing Machinery.

Kilian, J., Kushilevitz, E., Micali, S., and Ostrovsky, R. (1994). Reducibility and completeness in multi-party private computations. In *Symposium on Foundations of Computer Science*. IEEE.

Vaya, S. (2007). Secure multiparty computation on incomplete networks. ePrint archive version Mar'09.

Vaya, S. (2010). Brief Announcement: Realizing secure multiparty computation on incomplete networks. To appear in *Principles of Distributed Computing*. Association for Computing Machinery.

Yao, A. (1982). Protocols for secure computation. In *Symposium on Foundations of Computer Science*. IEEE.

APPENDIX: DISCUSSION OF RESULTS IN (Garay and Ostrovsky, 2008)

We consider the definition of privacy and the proof of main Theorem 4.3 in (Garay and Ostrovsky, 2008) in detail.

Definition of Privacy in (Garay and Ostrovsky, 2008). An input indistinguishability type definition of privacy is presented in (Garay and Ostrovsky, 2008). The definition has two components: (a) *Privacy of the Commitment phase* (b) *Privacy of the Computation phase of protocol*. For the computation phase the requirement is: As long as the committed input values of the sacrificed and corrupted parties and the computed output value are same, the distribution of views of the adversary (which is the vector of views of the corrupted parties generated from the Computation phase) are indistinguishable. (b) For the commitment phase, it is required that as long as initial input values and committed input values of the sacrificed and corrupted parties are same, the distribution of the views of the adversary generated from the commitment phase of the protocol are indistinguishable for the different scenarios in which rest of the honest parties start with different initial input values.

Protocol Π Tealizing the Definition of Security. Let G_n be a graph that has the requisite infrastructure to conduct a.e.s.c. i.e., is almost everywhere admissible graph. The protocol for a.e.s.c. on network G_n is reproduced verbatim from (Garay and Ostrovsky, 2008):

> First, we specify the communication structure of the simulation. Each round of protocol *C&C* MPC for complete networks is thought of as a 'super-round'. Each super-round has the same structure, with players taking turns (in, say, lexicographic order) to perform the simulation of sends and receives required in the original round. More specifically, at the onset, each player locally invokes procedure SELECT-PATH[1](G_n, P_i, P_j) computable map given by G_n, to obtain the set PATHS(P_i, P_j), for every P_i and P_j. Whenever P_i is required to send message m to P_j, P_i and P_j run PUB-SMT[2]$(P_i, P_j, m, PATHS(P_i, P_j))$; Similarly an invocation by P_i to broadcast in $(n, \frac{n}{3})$-VSS protocol is replaced by an invocation to the almost everywhere broadcast protocol with P_i as the sender.

Proof that Protocol Π Realizes the Definition of Privacy, Theorem 4.3, (Garay and Ostrovsky, 2008). The argument for the proof of Theorem 4.3 (that above protocol realizes the definition of security) in (Garay and Ostrovsky, 2008) is presented verbatim: (1) For privacy of computation phase:

> Privacy of the computation phase follows from a hybrid argument and reduction to the privacy of the message transmission scheme. In a fully connected network, the condition of indistinguishable views for the adversary for all $\overrightarrow{X^*_{V-T}}$, $\overrightarrow{Y^*_{V-T}}$ and $\overrightarrow{Z^*_T}$ such that the output of the function is the same, i.e., $f(\overrightarrow{X^*_{V-T}}, \overrightarrow{Z^*_T}) = f(\overrightarrow{Y^*_{V-T}}, \overrightarrow{Z^*_T})$ (Here, $\overrightarrow{X}^*_{V-T}$ and $\overrightarrow{Y}^*_{V-T}$ refers to the vector of input values of unsacrificed honest parties and $\overrightarrow{Z}^*_T$ refers to the vector of input values of corrupted and sacrificed honest parties.) is known to hold for an information-theoretically secure MPC protocol as long as the corrupted sets are the same (BenOr et al., 1988). Thus, if the adversary is able to distinguish the two views with non-negligible

[1]The procedure SELECT-PATH enumerates a set of paths in graph G_n between parties P_i and P_j which are used for sending messages by the protocol

[2]Protocol PUB-SMT is the protocol for perfect secure message transmission to be executed between parties P_i and P_j

> advantage in the simulated execution, then there would be a particular super-round in turn, player turn; in turn, message transmission where the adversary can distinguish the two runs on G_n but does not distinguish them in the fully connected network. This, in turn, contradicts the security of the message transmission protocol between two privileged players.

(2) The argument given in (Garay and Ostrovsky, 2008) to establish privacy of the input commitment phase is:

> The privacy of these values for players in W, follows from the privacy condition of $PUB-SMT$, which again these players are able to execute successfully and which guarantees that the views of the adversary (as well as other honest players, since the graph is $(2,t)$-admissible) under the transmission of any two messages are identical.

Problem with the Argument Considered Verbatim. Note that the "adversary" is just an Interactive Turing machine. Thus, the above argument for privacy of computation phase, is making claim about some adversary program "T" attacking simulated execution on G_n and execution of the vanilla BGW protocol on the fully connected network. However, it is easy to construct an adversary program "T" which works in a manner such that there is no "meaningful" correlation or correspondence between (distribution of) T's view(s) generated from simulation on network G_n (as described above) and (distribution of) T's view(s) generated from execution of vanilla BGW protocol on fully connected network. For example, consider the following ITM "T": If "T" is participating in the execution of simulation over network G_n, then "T" is programmed so that the corrupted parties behave arbitrarily (for example, don't participate in the protocol at all or share bad secrets during the execution of Verifiable secret sharing (VSS) protocol, Or send corrupted messages of certain type etc.). However, if "T" participates in the execution of (the appropriate) protocol over the fully connected network, then the corrupted parties execute protocols for honest parties. In fact, these views of the adversary from the two scenarios are semantically so different, that the indistinguishability of views from two runs of former type cannot be used to make any meaningful reduction to or inference about (in)distinguishability of views of latter type. It is possible that this problem arises due to brevity of presentation. The point to emphasize is that at an outer level reduction argument is necessary.

Conceptual Improvement to the Definition of Privacy

We observe the following conceptual improvement to the Input indistinguishability type definition of privacy given in (Garay and Ostrovsky, 2008). The authors define privacy of commitment phase and computation phase separately and combine them to conclude privacy of the full protocol. In other words, let X_1, Y_1 be the random variables denoting the views of adversary generated in Phase I (commitment) and Phase II (computation) of multiparty protocol for scenario (a) (appropriately quantified by input/output values). Similarly, X_2, Y_2 respectively denote the random variable denoting the views of adversary for scenario (b). The input indistinguishability type definition in GO08, specifies $X_1 \approx X_2$ and $Y_1 \approx Y_2$. The privacy definition for unconditional multiparty protocol requires the following condition to hold true: $(X_1, Y_1) \approx (X_2, Y_2)$. This latter correct requirement does not follow from the former. For example, let $X_1 = r_1, Y_1 = r_2$, $X_2 = r_1$ and $Y_2 = r_1$, where r_1, r_2 are distributed according to uniform probability distribution. Then, obviously $X_1 \approx Y_1$ and $X_2 \approx Y_2$, but $(X_1, X_2) \not\approx (Y_1, Y_2)$. In other words, definition of privacy must *properly* take into account correlation of views of adversary generated from the two phases of the protocol. The simulator based definition of privacy given in (Vaya, 2007) implies such a composite definition.