On Virtual Grey Box Obfuscation for General Circuits

Nir Bitansky[1,⋆], Ran Canetti[1,2,⋆⋆], Yael Tauman Kalai[3], and Omer Paneth[2,⋆⋆⋆]

[1] Tel Aviv University, Tel Aviv, Israel
[2] Boston University, Boston, MA, USA
[3] Microsoft Research New England, Cambridge, MA, USA

Abstract. An obfuscator $\mathcal{O}$ is Virtual Grey Box (VGB) for a class $\mathcal{C}$ of circuits if, for any $C \in \mathcal{C}$ and any predicate π, deducing $\pi(C)$ given $\mathcal{O}(C)$ is tantamount to deducing $\pi(C)$ given unbounded computational resources and polynomially many oracle queries to C. VGB obfuscation is often significantly more meaningful than indistinguishability obfuscation (IO). In fact, for some circuit families of interest VGB is equivalent to full-fledged Virtual Black Box obfuscation.

We investigate the feasibility of obtaining VGB obfuscation for general circuits. We first formulate a natural strengthening of IO, called *strong IO* (SIO). Essentially, $\mathcal{O}$ is SIO for class $\mathcal{C}$ if $\mathcal{O}(C) \approx \mathcal{O}(C')$ whenever the pair (C, C') is taken from a distribution over $\mathcal{C}$ where, for all x, $C(x) \neq C'(x)$ only with negligible probability.

We then show that an obfuscator is VGB for a class $\mathcal{C}$ if and only if it is SIO for $\mathcal{C}$. This result is unconditional and holds for any $\mathcal{C}$. We also show that for some circuit collections, SIO implies virtual black-box obfuscation.

Finally, we formulate a slightly stronger variant of the semantic security property of graded encoding schemes [Pass-Seth-Telang Crypto 14], and show that existing obfuscators such as the obfuscator of Barak et. al [Eurocrypt 14] are SIO for all circuits in NC^1, assuming that the underlying graded encoding scheme satisfies our variant of semantic security.

Put together, we obtain VGB obfuscation for all NC^1 circuits under assumptions that are almost the same as those used by Pass et. al to obtain IO for NC^1 circuits. We also show that semantic security is in essence *necessary* for showing VGB obfuscation.

1 Introduction

Program obfuscation, namely the ability to efficiently compile a given program into a functionally equivalent program that is "unintelligible", is an intriguing

⋆ Supported by an IBM Ph.D. Fellowship, the Check Point Institute for Information Security, and the Israeli Ministry of Science and Technology.

⋆⋆ Supported by the Check Point Institute for Information Security, an NSF EAGER grant, and NSF Algorithmic Foundations grant no. 1218461.

⋆⋆⋆ Supported by the Simons award for graduate students in theoretical computer science and NSF Algorithmic Foundations grant no. 1218461.

J.A. Garay and R. Gennaro (Eds.): CRYPTO 2014, Part II, LNCS 8617, pp. 108–125, 2014.

concept. Indeed, much effort has been devoted to understanding this concept from the definitional aspect, the algorithmic aspect, and the applications aspect. Here let us concentrate on the first two aspects.

Starting with the works of Hada [Had00] and Barak et al. [BGI+01], a number of measures of security for program obfuscation have been proposed. Let us briefly review three notions of interest. The first, *virtual black box (VBB)* obfuscation [BGI+01], requires that having access to the obfuscated program is essentially the same as having access to the program only as black box. Concretely, focusing on programs represented as circuits, an obfuscator $\mathcal{O}$ for a family of circuits is worst-case VBB if for any poly-time adversary $\mathcal{A}$, there exists a poly-time simulator S, such that for any circuit C from the family, and any predicate $\pi(\cdot)$, $\mathcal{A}$ cannot learn $\pi(C)$ from $\mathcal{O}(C)$ with noticeably higher probability than $\mathcal{S}$ can, given only oracle access to C. The obfuscator $\mathcal{O}$ is average-case VBB if the above is only required to hold for circuits C that are sampled at random from the family.

While this VBB obfuscation is natural and strong, Barak et al. [BGI+01] showed that this definition, and variants thereof, are unobtainable in general by demonstrating a family of *unobfuscatable functions:* these are functions f where any circuit computing the function inherently leaks secrets that are infeasible to compute given only black box access to f. Moreover it turns out that, under cryptographic assumptions, if the simulator $\mathcal{S}$ is universal (or equivalently, works for any adversarial auxiliary input) then VBB obfuscation is unobtainable for *any* circuit family whose functionality has super-polynomial "pseudo entropy" [GK05, BCC+14].

A weaker variant of VBB, called *virtual grey-box (VGB)* [BC10], allows the simulator to be *semi-bounded,* namely it can be computationally unbounded, while still making only a polynomial number of queries to the circuit C. While significantly weaker than VBB in general, VGB is still meaningful for circuits that are unlearnable even by semi-bounded learners. Furthermore, VGB obfuscators for circuits escape the general impossibility results that apply to VBB obfuscators.

A weaker notion yet, called indistinguishability obfuscation (IO) [BGI+01], allows the (now computationally unbounded) simulator to also make an unbounded number of queries to C. Equivalently, $\mathcal{O}$ is an IO for a circuit collection if for any two circuits C_0 and C_1 in the collection, having the same size and functionality, $\mathcal{O}(C_0)$ and $\mathcal{O}(C_1)$ are indistinguishable.

While IO has some attractive properties (e.g., any IO is the "best possible" obfuscation for its class), and some important cryptographic applications [SW13, GGH+13b], the security guarantees provided by IO are significantly weaker than those provided by either VBB or VGB obfuscation.

On the algorithmic level, for many years we had candidate obfuscators only for very simple functions such as point functions and variants. The landscape has changed completely with the recent breakthrough work of [GGH+13b], which proposed a candidate general-purpose obfuscation algorithm for all circuits.

[GGH+13b] show that their scheme resists some simple attacks; but beyond that, they do not provide any analytic evidence for security.

Considerable efforts have been made to analyze the security of the [GGH+13b] obfuscator and variants. The difficulty appears to be in capturing the security properties required from the *graded encodings schemes* [GGH13a, CLT13], which is a central component in the construction. As a first step towards understanding the security of the [GGH+13b] obfuscator, [BR13, BGK+13] consider an ideal algebraic model, where the adversary is given "generic graded encodings" that can only be manipulated via admissible algebraic operations. They show that, in this model, variants of the [GGH+13b] scheme are VBB obfuscators for all poly-size circuits. (We remark that [CV13] construct a VBB general obfuscator with similar properties; however their abstract model is different and does not seem to correspond to any existing cryptographic primitive.)

Still, neither of these idealized constructions or their analyses have, in of themselves, any bearing on the security of obfuscation algorithms in the plain model.

Pass et al. [PTS13] make the first step towards proving the security of a general obfuscation scheme based on some natural hardness assumption in the plain model. Specifically, they define a *semantic security* property for graded encoding schemes, which is aimed at capturing what it means for a graded encoding scheme to "behave essentially as an ideal multi-linear graded encoding oracle". They then show, assuming the existence of such a semantically-secure encoding scheme, that a specially-crafted variant of the [BGK+13] obfuscator, with the graded encoding scheme replaced by a semantically-secure graded encoding scheme, is IO for all circuits.

In this work we address the following question:

> *What is the strongest form of security for general obfuscation that can be based on natural cryptographic assumptions such as semantically-secure graded encoding?*

Our contributions. As our main result we obtain worst-case VGB obfuscation for NC^1, based on almost the same assumptions as those used in [PTS13] to show IO for NC^1. As an intermediate step towards this goal, we put forth a somewhat stronger variant of indistinguishability obfuscation, called *strong IO* (SIO). Informally, an obfuscator $\mathcal{O}$ is SIO for a class of circuits $\mathcal{C}$ if $\mathcal{O}(C) \approx \mathcal{O}(C')$ not only when $C, C' \in \mathcal{C}$ have the same functionality, but also when C and C' come from distributions over circuits in $\mathcal{C}$ that are "close together", in the sense that at any given input x, the probability that $C(x) \neq C'(x)$ is negligible. An alternative view of the definition (which turns out to be equivalent) is that if no adversary (even computationally unbounded) can distinguish oracle access to C from access to C' given only polynomial many queries, then the obfuscated circuits should be indistinguishable as well.

We then show that:

1. Strong IO is in fact *equivalent* to worst-case VGB obfuscation. Furthermore, for certain classes of functions, such as point functions, hyperplanes, or fuzzy

point functions, SIO is equivalent to full-fledged worst-case VBB obfuscation. These equivalences hold unconditionally. We consider this to be the main technical step in this work.

2. Assuming the existence of graded encoding schemes that satisfy a somewhat stronger variant of the semantic security notion of Pass et al. [PTS13], we show that known obfuscation schemes are SIO for all circuits in NC^1. More generally, we show that *any* obfuscator for a class of circuits $\mathcal{C}$ that is VBB in the ideal graded encoding model, is SIO in the plain model, when the ideal graded encoding oracle is replaced by a graded encoding scheme that satisfies a variant of the [PTS13] assumption.

We also give evidence for the *necessity* of semantically-secure graded encoding for obtaining VGB. Specifically we show that, assuming the existence of VGB obfuscators for all circuits, there exists *mutlilinear jigsaw puzzles*, a simplified variant of multilinear maps [GGH$^+$13b], that satisfies a form of semantic security. Such mutlilinear jigsaw puzzles are sufficient for obtaining the positive result described in Item 2 above.

Finally, we investigate the plausibility of the semantic security assumption on graded encoding schemes, propose some relaxed variants, and show that our main results can be obtained under all these relaxations. Namely, we first give new evidence for the relative strength of the semantically-secure graded encodings assumption. Specifically, we show that semantically-secure graded encodings are subject to the following limitations:

1. *SAT lower bounds.* We show that semantically-secure graded encodings imply exponential circuit lower bounds for SAT. Such lower bounds are currently not known to follow from IO (even assuming $P \neq NP$).
2. *A generic attack.* We present an attack showing that any graded encoding scheme with certain efficiency properties cannot satisfy semantic security. While the attack does not apply to currently known candidate graded encodings [GGH13a, CLT13], it does point out potential limitations of this notion. We complement this observation by suggesting a natural relaxation of semantic security called *bounded semantic-security* that bypasses this attack. Our main results can be obtained also under this relaxed assumption.

In addition to the above relaxation, we consider several other relaxations, and investigate their relations. We show that our main results can be obtained under all these relaxations.

The rest of the introduction provides a more detailed overview of our results. Section 1.1 presents the implication from SIO to VGB and VBB obfuscation. Section 1.2 provides background on graded encoding schemes and the semantic security assumption. Sections 1.3 presents the construction of strong IO obfuscators from semantically-secure graded encoding schemes, and Section 1.4 describes additional results on the viability of the semantic security assumption of graded encoding schemes, and relations among various variants.

1.1 From Strong IO to VGB and VBB Obfuscation

We first define strong IO a bit more precisely. A distribution $\tilde{C}$ over circuits is said to be ε-*concentrated* around a boolean function f if for any value x in the domain of f we have that $\Pr[\tilde{C}(x) \neq f(x)] \leq \varepsilon$. We say that $\tilde{C}$ is simply *concentrated* if it is ε-concentrated for some negligible function ε. An obfuscator $\mathcal{O}$ is strong IO for a class $\mathcal{C}$ of circuits if $\mathcal{O}(\tilde{C}) \approx \mathcal{O}(\tilde{C}')$ for any two distributions $\tilde{C}, \tilde{C}'$ over circuits in $\mathcal{C}$ that are concentrated around the same function. We stress that these distributions need not be efficiently samplable. We show the following.

Theorem 1 (informal). *An obfuscator is SIO for a class of circuits $\mathcal{C}$ if and only if it is worst-case VGB obfuscator for $\mathcal{C}$.*

Showing that VGB implies SIO is straightforward. In the other, more challenging, direction we construct an (inefficient) simulator $\mathcal{S}$ for any adversary $\mathcal{A}$. Recall that, for *any* circuit $C \in \mathcal{C}$ in the given collection $\mathcal{C}$, the simulator $\mathcal{S}$ should simulate what $\mathcal{A}$ learns from an obfuscation $\mathcal{O}(C)$, given only oracle access to C. The high level idea is as follows: $\mathcal{S}$ will use its oracle to C to gradually reduce the set $\mathcal{K}$ of candidates for the circuit C, starting from $\mathcal{K}_0 = \mathcal{C}$, and ending with a smaller set of candidates

$$\mathcal{K}_i \subsetneq \mathcal{K}_{i-1} \subsetneq \cdots \subsetneq \mathcal{K}_0 = \mathcal{C}.$$

$\mathcal{S}$ will continue this process until it obtains a set $\mathcal{K}^*$ where $\mathcal{A}$ cannot distinguish an obfuscation $\mathcal{O}(C)$ of C from an obfuscation $\mathcal{O}(C')$ of a random circuit C' in $\mathcal{K}^*$.

To carry out this plan, $\mathcal{S}^C$ iteratively performs two main steps: *concentration*, and *majority separation*. In the concentration step $\mathcal{S}$ tries to learn C in a straightforward way: it queries C on a point x_i that splits the current set of candidate circuits $\mathcal{K}_i$ as evenly as possible. Based on the value of $C(x_i)$, $\mathcal{S}$ rules out some of the candidates. This process is repeated until there is no query that necessarily shrinks the set of candidates by a factor of at least $1 - \varepsilon$, where ε is a parameter of the simulation that is chosen such that $1/\varepsilon$ is a polynomial, depending only on $\mathcal{A}$ and on the required simulation accuracy. Note that at the end of the concentration step, $\mathcal{S}$ must reach a set of candidates $\mathcal{K}_j$ that is ε-concentrated. This occurs after at most $\log|\mathcal{C}|/\varepsilon$ queries. The concentration step alone essentially suffices to ensure *average-case* VGB simulation; indeed, we show that for a circuit C chosen at random from a concentrated set $\mathcal{K}_j$, $\mathcal{A}$ cannot compute any predicate $\pi(C)$, given $\mathcal{O}(C)$, better than it can given an obfuscation $\mathcal{O}(C')$ of an independent random $\mathcal{C}' \leftarrow \mathcal{K}_j$.

However, the concentration step alone does not guarantee *worst-case* simulation. In particular, $\mathcal{A}$ may have some hardwired information that allows it to distinguish C from a random circuit in $\mathcal{K}_j$. In this case, however, $\mathcal{S}$ can further reduce the set of candidates $\mathcal{K}_j$ by separating any such distinguishable circuit C from the majority $\mathrm{maj}_{\mathcal{K}_j}$. Concretely, we define the set $\mathbb{D}_{\mathcal{A}}(\mathcal{K}_j)$ of *distinguishable circuits* in $\mathcal{K}_j$, as those circuits C in $\mathcal{K}_j$ such that $\mathcal{A}$ can ε-distinguish between $\mathcal{O}(C)$ and $\mathcal{O}(C')$ for a random $C' \leftarrow \mathcal{K}_j$.

In the majority-separation step, the simulator will query its oracle C on a *small* set of roughly $\log|\mathcal{C}|/\varepsilon$ points $L_{\mathcal{K}_j}$ that *separates* all the distinguishable circuits in $\mathbb{D}_{\mathcal{A}}(\mathcal{K}_j)$ from the majority circuit $\mathrm{maj}_{\mathcal{K}_j}$. This means that, if the oracle C agrees with the majority $\mathrm{maj}_{\mathcal{K}_j}$ on all points $x \in \mathcal{L}_{\mathcal{K}_j}$, then $\mathcal{A}$ cannot tell apart $\mathcal{O}(C)$ form $\mathcal{O}(C')$ for a random $C' \leftarrow \mathcal{K}_j$, in which case, the simulation can be completed. Otherwise, $\mathcal{S}$ manages to separate C from $\mathrm{maj}_{\mathcal{K}_j}$, and obtain a new set of candidates $\mathcal{K}_{j+1} \subsetneq \mathcal{K}_j$ which is necessarily smaller by a $1-\varepsilon$ factor, since $\mathcal{K}_j$ is ε-concentrated.

By iteratively applying the two steps we either reach some $\mathcal{K}^*$ for which $\mathcal{A}$ cannot distinguish $\mathcal{O}(C)$ from $\mathcal{O}(C')$ for a random $C' \leftarrow \mathcal{K}^*$, or we have completely exhausted the collection $\mathcal{C}$ and found exactly the circuit C. In any case, since we reduce $\mathcal{K}_j$ at each step by a $1-\varepsilon$ factor, the process must end after at most $\log|\mathcal{C}|/\varepsilon$ steps, and at most $\mathrm{poly}(\log|\mathcal{C}|/\varepsilon)$ queries.

But how do we establish the existence of a small set $L_{\mathcal{K}_j}$ that separates $\mathbb{D}_{\mathcal{A}}(\mathcal{K}_j)$ from the majority $\mathrm{maj}_{\mathcal{K}_j}$? Here we rely on the fact that $\mathcal{O}$ is a strong IO obfuscator. Specifically, strong IO implies that any subset S of the distinguishable circuits $\mathbb{D}_{\mathcal{A}}(\mathcal{K}_j)$, cannot be ε-concentrated around $\mathrm{maj}_{\mathcal{K}_j}$, because $\mathcal{A}$ distinguishes $\mathcal{O}(C)$, for $C \leftarrow \mathcal{K}_j$ from $\mathcal{O}(C')$ for $C' \leftarrow S \subseteq \mathbb{D}_{\mathcal{A}}(\mathcal{K}_j)$.[1] Since no S as above is ε-concentrated around $\mathrm{maj}_{\mathcal{K}_j}$, we can separate all of the circuits in $\mathbb{D}_{\mathcal{A}}(\mathcal{K}_j)$ from $\mathrm{maj}_{\mathcal{K}_j}$ with at most $\mathrm{poly}(\log|\mathcal{C}|/\varepsilon)$ points, as required.

On the possibility of VBB obfuscation. The simulation strategy described above requires only a polynomial number of queries $\mathrm{poly}(\log|\mathcal{C}|/\varepsilon)$; however, the overall running time of the simulator may not be bounded in general. Indeed, in the concentration step, finding a point x_j that significantly splits $\mathcal{K}_j$ may require super-polynomial time. Also, in the majority-separation step, while the sets $L_{\mathcal{K}_j}$ are small, computing them from $\mathcal{K}_j$ may also require super-poly time.

Nevertheless, we show that for certain classes of circuits, simulation can be done more efficiently, or even in polynomial time. Specifically, abstracting away from the above simulation process, we consider the notion of *learning via a majority-separation oracle*, where a given circuit C (or more generally a function) in a prescribed family is learned via oracle access to C and oracle access to the majority separation oracle $\mathbb{S}$, which takes as input (the description of a) a concentrated sub-family $\mathcal{K}$ that includes C and outputs a point x that separates C from $\mathrm{maj}_{\mathcal{K}}$ (the majority of functions in $\mathcal{K}$).

The complexity of our simulator is then determined by how well can the class in question be learned by majority-separation oracles. While the strategy described above shows that any class of circuits can be learned by a majority-separation oracle with polynomially many queries to C and $\mathbb{S}$, the pattern of these queries and the way in which they are interleaved affects the complexity of the simulation. As a simple example, suppose that there is a constant number of oracle calls to either $\mathbb{S}$ or C. (This is the case in some classes for which worst-case VBB obfuscation was previously shown, such as point functions, constant-size

[1] We assume here (for simplicity and without loss of generality), that the distinguishing gap is always of the same sign.

set functions, and constant dimension hyper-planes.) In this case we can non-uniformly hardwire in advance a polynomial number of separating sets $\mathcal{L}_{\mathcal{K}_j}$ into our simulator, without having to compute them on the fly. Otherwise, the sets $\mathcal{L}_{\mathcal{K}_j}$ are determined *adaptively* and need to be computed on the fly.

Comparison to [BBC+14]. Barak et al. show that average-case VBB obfuscation for all *evasive* collections (these are collections that are concentrated around the constant zero function) implies *weak average-case* VGB for all collections, where weak VGB means that the simulator is allowed to make a slightly super-polynomial number of queries. The result is weaker in the sense that it only achieves average-case (rather than worst-case) simulation, and only weak VGB.

At a technical level, what allows us to get standard VGB, as opposed to weak VGB, is the fact that we assume IO for the family in question. More specifically, the level to which the simulator has to concentrate the candidate set is determined by the adversary and simulation quality. In the time of obfuscation, these parameters are not known. Relying on IO allows us to push the decision of how many iterations to make all the way to the simulation rather than having to make this decision at the time of obfuscation.

1.2 Semantically-Secure Graded Encoding Schemes: Background

Before describing how we get strong IO from semantically secure graded encoding schemes, we provide some background on the latter. A graded encoding scheme [GGH13a] consists of the following algorithms: $\mathsf{InstGen}$ that give a universe set $[k]$, outputs public parameters pp and secret parameters sp, where pp contains a description of a ring R; Encode that given sp, a set $S \subseteq [k]$ and $\alpha \in R$, generates an encoding $[\alpha]_S$; Add and Sub that, given encodings $[\alpha_1]_S$ and $[\alpha_2]_S$, generate encodings $[\alpha_1 + \alpha_2]_S$ and $[\alpha_1 - \alpha_2]_S$ respectively; Mult that, given encodings $[\alpha_1]_{S_1}$ and $[\alpha_2]_{S_2}$ such that $S_1 \cap S_2 = \emptyset$, generates an encoding $[\alpha_1 \cdot \alpha_2]_{S_1 \cup S_2}$; and isZero that given an encoding $[\alpha]_{[k]}$ outputs 1 if and only if $\alpha = 0$ (all the algorithms above also take as input pp).

[GGH13a, CLT13] consider standard versions of DDH-type security that can be conjectured to hold for their graded encoding schemes. Basing the security of obfuscation mechanisms on these assumptions seems at this point far out of reach, even if one considers only IO security. So which security properties of encoding schemes would suffice for this purpose? The high-level approach of Pass et al. [PTS13] is to devise a property that, not only hides "DDH-type relations" between encodings, but also any other relation that cannot be revealed using the admissible algebraic operations provided by the graded encoding interface. In other words, the encoding scheme should amount to an "ideal encoding scheme", where encodings are truly accessed only through admissible algebraic operations. This may, in particular, allow leveraging the existing proofs of VBB security in the ideal graded encoding model [BR13, BGK+13].

More specifically, Pass et al. take the following approach (described first in an oversimplified manner). Consider a *message sampler* $\mathbb{M}([k], R)$ that samples, from one of two distributions $\mathcal{D}_0$ or $\mathcal{D}_1$, a tuple $(S_1, m_1), \ldots, (S_\ell, m_\ell)$, where

each $S_i \subseteq [k]$ and each $m_i \in R$, and ℓ is polynomial in the security parameter. We say that the sampler is *admissible* if no polynomially-bounded "algebraic adversary" that is given $\boldsymbol{S} = (S_1, \ldots, S_\ell)$, and can access the ring elements $\boldsymbol{m} = (m_1, \ldots, m_\ell)$ only via *an ideal encoding oracle*, is able to tell whether $(\boldsymbol{S}, \boldsymbol{m})$ were taken from $\mathcal{D}_0$ or $\mathcal{D}_1$. The ideal encoding oracle only allows the same algebraic manipulations allowed by the graded encoding interface, or put abstractly, it allows the adversary to choose any arithmetic circuit C that respects the set structure given by $\boldsymbol{S}$, and test whether $C(\boldsymbol{m}) = 0$. The requirement is that, for such an admissible sampler, an efficient adversary that obtains actual encodings $([m_i]_{S_i} : i \in [\ell])$, along with the corresponding public parameters pp, also cannot tell whether $(\boldsymbol{S}, \boldsymbol{m})$ was sampled from $\mathcal{D}_0$ or $\mathcal{D}_1$.

As noticed by Pass et al., the assumption formulated above is actually false—it is susceptible to a diagonalization attack in the spirit of the [BGI+01] impossibility result for general VBB obfuscation. More specifically, the unobfuscatable functions constructed in [BGI+01] can be directly used to obtain two distributions on circuits $\mathcal{C}_0$ and $\mathcal{C}_1$ which cannot be distinguished given only black-box access to C sampled from either $\mathcal{C}_0$ or $\mathcal{C}_1$, but given any circuit with the same functionality as the circuit C, it is easy to tell from which one of the two C was sampled from. This distinguishing attack could now be translated to our setting using any obfuscation scheme that uses ideal graded encoding, such as the ones of [BR13, BGK+13]. Indeed, we can define an admissible sampler that corresponds to distributions $\mathcal{D}_0$ and $\mathcal{D}_1$, which sample $(\boldsymbol{S}, \boldsymbol{m})$ by first sampling a circuit C taken from $\mathcal{C}_0$ or $\mathcal{C}_1$, respectively, and letting $(\boldsymbol{S}, \boldsymbol{m})$ correspond to the ideal obfuscation of C. Admissibility is guaranteed due to the VBB guarantee in the ideal encoding model, whereas in the real world, the actual encodings $([m_i]_{S_i})$ give a circuit that computes the same function of C, and thus allows determining from where the sample was taken.

Pass et al. get around this caveat by strengthening the admissibility requirement to require that $\mathcal{D}_0, \mathcal{D}_1$ are indistinguishable even to a *semi-bounded* algebraic adversary, namely an algebraic adversary that is computationally unbounded, but makes only a polynomial number of queries to the ideal graded encoding oracle. The above attack no longer applies since the circuit distributions $\mathcal{C}_0$ and $\mathcal{C}_1$ involve computational elements, such as encryption, which makes them completely distinguishable to unbounded attackers, even given only polynomially many oracle queries. More generally, as mentioned above, we do not have any analogous unobfuscatability results for VGB obfuscation, and thus there are no known attacks on this notion of semantic security.

Furthermore, Pass et al. show that even this relaxed assumption suffices for obtaining IO in the plain model. This is the case since for NC^1 circuits, the [BGK+13] obfuscator in the ideal-graded encoding model is VBB even against *semi-bounded* adversaries. (VBB, in this context, means that the simulator is poly-time, given oracle access to the algebraic adversary and the obfuscated program.) The eventual Pass et al. assumption is further relaxed in several ways, while still yielding their main application to IO.

1.3 Strong IO from Semantically-Secure Graded Encoding, and Back Again

We sketch our variant of the semantic security assumption, and explain how we obtain strong IO for NC^1 circuits from this variant. We also give evidence for the *necessity* of semantic security for obtaining strong IO.

To get strong IO for arbitrary circuit distributions (including distributions that are not efficiently samplable) we will need to rely on a somewhat stronger version of the semantic security assumption discussed above, that allows for computationally unbounded samplers. Some care has to be taken when formalizing this assumption.

Recall that the message sampler is given the description of a ring R. (This is required in order to sample obfuscations in the ideal graded encoding model that consist, for example, of random elements in R.) A computationally unbounded sampler that sees R may be able to recover information that compromises the security of the encodings (for example, the secret parameters). The sampler can produce encodings that reveal this secret information. Note that such a sampler may still be admissible since learning the secret parameters gives no advantage to an algebraic adversary. Luckily, however, we can do with a significantly weaker variant of semantic security where this attack is avoided.

Specifically, the sampling is done in two stages: first, an unbounded sampler S generates a poly-size auxiliary input string; second, an *efficient* encoder $\mathbb{M}$ gets the ring R and the auxiliary input string, and generates the final samples. We call this variant *strong-sampler semantic security.*

Strong-sampler semantic security is already sufficient for constructing strong IO for arbitrary circuit distributions. The idea is to have the unbounded sampler S sample the description of a circuit as an auxiliary input string, and then, the efficient encoder $\mathbb{M}$ samples an obfuscation of the auxiliary input circuit. We show, with a straightforward proof, that the following holds.

Theorem 2 (informal). *Let $\mathcal{O}$ be any obfuscator for a class $\mathcal{C}$ of circuits, that is VBB against semi-bounded adversaries in the ideal graded encoding model. Then instantiating the graded encoding oracle with a strong-sampler semantically-secure graded encoding scheme results in a strong indistinguishability obfuscator $\mathcal{O}'$ for $\mathcal{C}$, in the plain model.*

Then, relying on the Barak et al. obfuscation for NC^1 in the ideal graded encoding model [BGK+13] (which is indeed VBB against semi-bounded adversaries), we obtain the following corollary.

Corollary 1 (informal). *Assume there exists a strong-sampler semantically-secure encoding scheme. Then there exists a strong indistinguishability obfuscator for NC^1.*

We also give evidence for the *necessity* of semantically-secure graded encoding schemes for obtaining VGB. To this end, we focus on a version of graded encoding with restricted functionality called *mutlilinear jigsaw puzzles* [GGH+13b]. Unlike

graded encodings, in mutlilinear jigsaw puzzles, encodings can only be generated together the the system parameters. We refer to the public parameters, together with the set of initialized encodings, as a puzzle. Instead of performing individual permitted operations over the encodings, all the jigsaw puzzle user can do is to specify an arithmetic circuit C that respects the set structure of the set of initialized encodings, and test whether C evaluates to 0 on these encodings or not. Semantic security of mutlilinear jigsaw puzzles is defined similarly to the graded encoding case. Despite their restricted functionality, semantically-secure mutlilinear jigsaw puzzles can replace graded encodings in our construction of strong IO for NC^1.

We observe that the existence of semantically-secure jigsaw puzzles is implied by VGB obfuscation for all circuits. To see why this is the case, consider the circuit P that has a set of ring elements $\boldsymbol{m} = (m_1, \ldots, m_\ell)$ hardwired into it, together with the corresponding sets $\boldsymbol{S} = (S_1, \ldots, S_\ell)$. The circuit P takes as input an arithmetic circuit C that respects the set structure given by $\boldsymbol{S}$, and tests whether $C(\boldsymbol{m}) = 0$. To initialize a puzzle from a set of encodings $(\boldsymbol{S}, \boldsymbol{m})$ we simply VGB obfuscate the circuit P.

1.4 More on Semantic Security

Next we discuss our results pertaining to the study of semantic security of graded encoding schemes. The negative results discussed below hold even for the basic notion of semantic security, where the message sampler is of polysize.

SAT lower bounds. As additional evidence to the power of semantically-secure graded encodings, we observe that they imply that there do no exist SAT solvers that run in time $2^{o(n)} \cdot \mathrm{poly}(|C|)$, for a boolean circuit C with n input variables; namely, any worst-case SAT solver must be exponential in the number of variables. To show this lower bound, we rely on a result by Wee [Wee05], showing a similar lower from any point function obfuscation.

Efficiency limitations via a generic attack. We present an attack against any graded encoding scheme satisfying certain efficiency properties. Before specifying these efficiency properties, let us first describe the high-level idea behind the attack, from which they emerge.

Similarly to the attack described in Section 1.2, this attack is based on ideal graded encoding obfuscation. However, this attack holds even when admissibility is defined with respect to semi-bounded algebraic adversaries, rather than just bounded ones. More specifically, it relies on the fact that the [BGK+13] ideal obfuscation scheme is also VBB with respect to semi-bounded algebraic adversaries. Recall that this on its own is not enough to recover the attack from Section 1.2, since there is no general impossibility VGB obfuscation. The attack we describe now will indeed take a somewhat different approach, exploiting the particular interface of graded encoding schemes.

The idea is to construct two circuit distributions $\mathcal{C}_0, \mathcal{C}_1$, where any circuit $C_{b,r}$ sampled from $\mathcal{C}_b$ is associated with a random ring element $r \in R$. The circuit

$C_{b,r}$ reveals the hidden bit b only when given as input some public parameters pp and a proper encoding $[r]_{[k]}$ of the ring element r. The corresponding admissible sampler $\mathbb{M}$ would then sample from one of two distributions $\mathcal{D}_0, \mathcal{D}_1$, where sampling from $\mathcal{D}_b$ is done by sampling $C_{b,r}$ from $\mathcal{C}_b$, and outputting $(\boldsymbol{S}, \boldsymbol{m})$, that represents an ideal obfuscation of $C_{b,r}$, together with $([k], r)$. Intuitively, an ideal, even semi-bounded, algebraic adversary gains no more than oracle-access to $C_{b,r}$, together with the ability to evaluate low-degree arithmetic circuits on the random ring element r, thus it cannot learn the bit b.[2] In contrast, the real world distinguisher, which is given the public parameters pp and an actual encoding of r, can simply run the obfuscation on pp and the encoding of r, and learn b.

So what is needed to make the above attack applicable? First, since we only have ideal obfuscation against semi-bounded adversaries for NC^1, the circuit $C_{b,r}$ should be implementable in NC^1. Second, it is required that the size of the public parameters pp and the size of an encoding grows slower than k, the size of the universe for the sets that control the depth of allowed arithmetic computations. Indeed, in order to obfuscate $C_{b,r}$ in the ideal graded encoding model, it is required that the universe set $[k]$ is appropriately large (in particular, larger than the circuit's input). Thus, the public parameters and encoding received by $C_{b,r}$ as input must grow slower than k.[3]

Both of the above efficiency requirements are not satisfied by the candidate constructions of [GGH13a, CLT13] in their current forms. Indeed, for these schemes it is not known how to implement $C_{b,r}$ (or any procedure of equivalent effect) in NC^1. Also, in these schemes the size of the public parameters and encodings does grow with the maximal level k.

Still, it may be good to keep this attack in mind, pending potential improvements in the efficiency of obfuscation algorithms or graded encoding schemes. In fact, it appears prudent to weaken semantic security by requiring that it holds only given some a priory bound on either the level k, or on the number of elements ℓ output by an admissible sampler $\mathbb{M}$. This allows considering candidate schemes where certain parameters, such as the size of the ring (and induced size of encodings), are larger than these bound and are thus not susceptible to this type of attacks.

There may also exist certain tradeoffs in efficiency. For instance, in certain settings it may be more reasonable to let the size of parameters grow with ℓ, rather than with k. We define such a variant of *bounded semantic security*.

Relaxations of semantic security. As mentioned above, the notion of semantically-secure graded encodings lends itself to a number of variants along several axes. We study the relations among these relaxations and show that eventually they all

[2] This argument is a bit oversimplified; indeed, to argue that one cannot learn b given oracle access to r, we should also deal with the case that the adversary queries with improper public parameters and encodings. In the body, we deal with this by using a variant of the circuit $C_{b,r}$ that checks that any query corresponds to at most a small number of ring elements, and thus hits r only with negligible probability.

[3] We thank Rafael Pass for pointing this out.

suffice for obtaining the implications presented above for program obfuscation. Below we address two relaxations that were introduced by Pass et al. [PTS13] (and are already embedded into their main definition of semantic security).

First, Pass et al. consider *constant-message samplers* where the first $\ell - O(1)$ elements in the distributions $\mathcal{D}_0$ and $\mathcal{D}_1$ are required to be exactly the same, and are viewed as polynomially long "auxiliary-input" correlated to the last constant number of elements. (In later versions, Pass et al. limit the number of elements in the auxiliary distribution $\mathcal{Z}$ to a fixed polynomial, partially in light of the attacks described in this work.) Second, they strengthen the notion of admissibility where indistinguishability with respect to the algebraic adversary needs to hold in a strong *pointwise* sense. That is, for almost any two samples $(\boldsymbol{S}_0, \boldsymbol{m}_0), (\boldsymbol{S}_1, \boldsymbol{m}_1)$, taken from (potentially joint) distributions $(\mathcal{D}_0, \mathcal{D}_1)$, the algebraic adversary outputs the same bit when given $(\boldsymbol{S}_0, \boldsymbol{m}_0)$ and when given $(\boldsymbol{S}_1, \boldsymbol{m}_1)$. Finally, admissibility is further strengthened to only allow for "highly-entropic" samples. Indeed, this relaxation turns out to be essential in the context of the [GGH13a] graded encoding scheme (but not necessarily in the [CLT13] scheme).

From a technical perspective, the difference between the constant-message and multi-message definitions (in either the pointwise or non-pointwise case) is that the general transformation of Theorem 2 appears to require the seemingly stronger multi-message definition. In contrast, the specific construction of Pass et al. works even using only the single-message version.

We show that, in fact, the relaxed constant-message notion is equivalent to the multi-message notion. However, there are certain nuances to this equivalence:

- In the non-pointwise case, we show that even single-message (rather than constant-message) semantic-security implies multi-message semantic-security. Here the reduction essentially preserves the number of elements output by the attacker's message sampler. Specifically, any attacker against m-message semantic-security translates to an attacker against single-message semantic-security with auxiliary input of length m.
- In the pointwise case, this implication still holds, but with certain loss in parameters. Specifically, any attacker against m-message semantic-security with distinguishing advantage ϵ translates to an attacker against single-message semantic-security with auxiliary input of length $m \cdot \mathrm{poly}(n/\epsilon)$.

In conclusion, in the non-pointwise case, constant-message semantic-security does not constitute an actual relaxation, and would also lead to a generic construction of (strong) IO. In the pointwise case, however, the specific obfuscator of Pass et al. gives a quantitative security advantage compared with the generic construction. In particular, for **bounded** constant-message pointwise semantic-security, the generic transformation does not apply as far as we know, whereas the obfuscator of Pass et al. does. It would thus be interesting to come up with evidence as to whether moving to pointwise security amounts to a meaningful relaxation of the assumption.

Strong IO from new assumptions? Pass et al. also consider an alternative modification of semantic security, where instead of requiring indistinguishability with respect to *any* admissible sampler, it is only required for a *single* specific sampler. To get IO, however, they also require that indistinguishability holds against subexponential distinguishers. This assumption is incomparable to the assumption discussed here, and is not further studied in this work. Gentry et al. [GLSW14] recently constructed indistinguishability obfuscators based on a new assumption of a somewhat different flavor, regarding a more demanding variant of graded encoding. Whether these assumptions suffice for constructing strong indistinguishability obfuscators is an intriguing question.

Organization. Section 2 reviews the definitions of VBB, VGB and IO. Section 3 defines SIO and shows its equivalence to VBB for concentrated circuit distributions. Section 4 constructs worst case VGB and VBB obfsucators from strong IO. Section 5 constructs SIO from semantically-secure graded encoding schemes. The study of the semantically-secure graded encoding assumption appear in the full version of this work.

2 Obfuscation: VBB, VGB, Indistinguishability

We review three basic definitions of obfuscation that are used throughout the paper. We start by defining the functionality requirement, which all the notions share, and then define different security notions.

Definition 1 (Functionality). *A PPT algorithm $\mathcal{O}$ is an obfuscator for a collection of circuits $\mathcal{C} = \bigcup_{n\in\mathbb{N}} \mathcal{C}_n$, if for any $C \in \mathcal{C}$,*

$$\Pr_{\mathcal{O}}[\forall x : \mathcal{O}(C)(x) = C(x)] = 1 \ .$$

VBB and VGB Obfuscation. Virtual Black Box (VBB) obfuscation [BGI+01] guarantees that an obfuscated circuit $\mathcal{O}(C)$ does not reveal any predicate $\pi(C)$ that cannot be learned by an efficient simulator that is given only black-box access to C. The basic definition is *worst-case* in the sense that the simulator needs to be successful for any circuit in a given circuit collection. We later also address an *average-case* notion. In the definition below we use a slightly weaker definition than the standard one, and allow the simulator to depend on the distinguishing probability p.

Definition 2 (Worst-case VBB Obfuscation). *An obfuscator $\mathcal{O}$ for a collection of circuits $\mathcal{C} = \bigcup_{n\in\mathbb{N}} \mathcal{C}_n$ is worst-case VBB if for every poly-size adversary $\mathcal{A}$, and polynomial p, there exists a poly-size simulator $\mathcal{S}$, such that for every $n \in \mathbb{N}$, every predicate $\pi : \mathcal{C}_n \to \{0,1\}$, and every $C \in \mathcal{C}_n$:*

$$\left| \Pr_{\mathcal{A},\mathcal{O}}[\mathcal{A}(\mathcal{O}(C)) = \pi(C)] - \Pr_{\mathcal{S}}[\mathcal{S}^C(1^n) = \pi(C)] \right| \leq 1/p(n) \ .$$

Virtual Grey Box (VGB) obfuscation [BC10] relaxes VBB by allowing the simulator to have unbounded computational power, but still only a bounded number of oracle queries to C.

Definition 3 (Worst-case VGB Obfuscation). *An obfuscator $\mathcal{O}$ for a collection of circuits $\mathcal{C} = \bigcup_{n\in\mathbb{N}} \mathcal{C}_n$ is worst-case VGB if for every poly-size adversary $\mathcal{A}$, and polynomial p, there exists an unbounded simulator $\mathcal{S}$, and a polynomial q, such that for every $n \in \mathbb{N}$, every predicate $\pi : \mathcal{C}_n \to \{0,1\}$, and $C \in \mathcal{C}_n$:*

$$\left| \Pr_{\mathcal{A},\mathcal{O}}[\mathcal{A}(\mathcal{O}(C)) = \pi(C)] - \Pr_{\mathcal{S}}[\mathcal{S}^{C[q(n)]}(1^n) = \pi(C)] \right| \leq 1/p(n) \ ,$$

where $C[q(n)]$ is an oracle that allows at most $q(n)$ queries.

Indistinguishability Obfuscation. We next define the notion of indistinguishability obfuscation, introduced in [BGI+01].

Definition 4 (Indistinguishability obfuscation [BGI+01]). *An obfuscator for $\mathcal{C}$ is said to be an* indistinguishability obfuscator *for $\mathcal{C}$, denoted by $i\mathcal{O}$, if for any poly-size distinguisher $\mathcal{D}$, there exists a negligible function μ such that for all $n \in \mathbb{N}$, and any two circuits $C_0, C_1 \in \mathcal{C}_n$ of the same size and functionality,*

$$\Pr[b \leftarrow \{0,1\}; \mathcal{D}(C_0, C_1, i\mathcal{O}(C_b)) = b] \leq \frac{1}{2} + \mu(n) \ .$$

It can be readily seen that if an obfuscator $\mathcal{O}$ is VBB for a function collection $\mathcal{C}$ then it is also VGB for $\mathcal{C}$. Furthermore, if $\mathcal{O}$ is VGB for $\mathcal{C}$ then it is also an indistinguishability obfuscator for $\mathcal{C}$.

3 Strong Indistinguishability Obfuscation

In this section we define the notion of strong indistinguishability obfuscation (SIO). We start by defining the notion of concentrated distributions over circuits.

Concentrated Circuit Distributions. At a high-level, a distribution ensemble $\tilde{\mathcal{C}}$, over a circuit collection $\mathcal{C}$, is *concentrated*, if given polynomially many oracle queries to a random circuit C from the distribution, it is information theoretically hard to find an input x such that C does not agree with $\mathrm{maj}_{\tilde{\mathcal{C}}}$ on the point x, where $\mathrm{maj}_{\tilde{\mathcal{C}}}$ is the common output of circuits distributed according to $\tilde{\mathcal{C}}$. If $\tilde{\mathcal{C}}$ corresponds to the uniform distribution on some collection $\mathcal{C}$, $\mathrm{maj}_{\tilde{\mathcal{C}}}$ is simply the majority vote. Concentrated distributions naturally generalize the concept of *evasive distributions* studied in [BBC+14], in which the majority is always the all-zero function, i.e. $\mathrm{maj}_{\tilde{\mathcal{C}}} \equiv 0$.

Definition 5 (Concentrated circuit distributions)
Let $\mathcal{C} = \bigcup_{n\in\mathbb{N}} \mathcal{C}_n$ be a circuit collection, where $\mathcal{C}_n$ consists of circuits $C : \{0,1\}^n \to \{0,1\}$ of size $\mathrm{poly}(n)$, and let $\tilde{\mathcal{C}}_n$ be a distribution on $\mathcal{C}_n$. Let $\mathrm{maj}_{\tilde{\mathcal{C}}_n}(x) := \lfloor \mathbb{E}_{C\leftarrow\tilde{\mathcal{C}}_n} C(x) \rceil$ be the common output at point x of circuits drawn from $\tilde{\mathcal{C}}_n$.

1. *For any* $\varepsilon \in [0,1]$, $\tilde{\mathcal{C}}_n$ *is said to be* ε*-concentrated if*

$$\max_{x\in\{0,1\}^n} \Pr_{C\leftarrow\tilde{\mathcal{C}}_n}\left[C(x) \neq \mathrm{maj}_{\mathcal{C}_n}(x)\right] \leq \varepsilon \ .$$

2. $\tilde{\mathcal{C}}$ *is said to be concentrated if for some negligible* $\mu(\cdot)$*, and any* $n \in \mathbb{N}$, $\tilde{\mathcal{C}}_n$ *is* $\mu(n)$*-concentrated.*
3. $\tilde{\mathcal{C}}$ *is said to be evasive if it is concentrated, and for any* $n \in \mathbb{N}$ *and any* $x \in \{0,1\}^n$, $\mathrm{maj}_{\tilde{\mathcal{C}}_n}(x) = 0$.
4. *We say that the collection* $\mathcal{C}$ *itself is concentrated (evasive) if the uniform distribution ensemble on circuits in* $\mathcal{C}$ *is concentrated (evasive).*

Strong Indistinguishability Obfuscation. Strong Indistinguishability Obfuscation requires that indistinguishability holds, even when C_0 and C_1 do not necessarily compute the exact same function, but are taken from two distributions $\tilde{\mathcal{C}}^0_n$ and $\tilde{\mathcal{C}}^1_n$ that are concentrated around the same function; namely, $\mathrm{maj}_{\tilde{\mathcal{C}}^0_n} \equiv \mathrm{maj}_{\tilde{\mathcal{C}}^1_n}$:

Definition 6 (Strong indistinguishability obfuscation). *An obfuscator for* $\mathcal{C}$ *is said to be a* **strong** indistinguishability obfuscator *for* $\mathcal{C}$*, denoted by* $i\mathcal{O}^*$*, if for any two concentrated distribution ensembles* $\tilde{\mathcal{C}}^0, \tilde{\mathcal{C}}^1$ *on* $\mathcal{C}$*, such that* $\forall n \in \mathbb{N} : \mathrm{maj}_{\tilde{\mathcal{C}}^0_n} \equiv \mathrm{maj}_{\tilde{\mathcal{C}}^1_n}$*, and any poly-size distinguisher* $\mathcal{D}$*, there exists a negligible function* μ *such that for all* $n \in \mathbb{N}$*,*

$$\Pr[b \leftarrow \{0,1\}; (C_0, C_1) \leftarrow (\tilde{\mathcal{C}}^0_n, \tilde{\mathcal{C}}^1_n); \mathcal{D}(i\mathcal{O}^*(C_b)) = b] \leq \frac{1}{2} + \mu(n) \ .$$

Remark 1. Above, we do not require that the distributions $\tilde{\mathcal{C}}^0, \tilde{\mathcal{C}}^1$ are efficiently samplable. We can also consider a weaker definition where this restriction is added. In the full version of this work we show that this weaker version can be obtained from a weaker notion of semantic security.

We observe that any strong IO obfuscator for $\mathcal{C}$ is also an IO obfuscator for $\mathcal{C}$. Indeed, for any two circuits C_0, C_1 of equivalent functionality, each of these circuits on its own is trivially concentrated around their common functionality.

4 Strong IO Is Equivalent to Worst-Case VGB

In this section, we prove that the notion of strong indistinguishability obfuscation (strong IO) is equivalent to VGB. Clearly, any VGB obfuscator for a class $\mathcal{C}$ is also a strong IO for $\mathcal{C}$. We show that the converse is true as well. Namely, we show that any strong indistinguishability obfuscator $\mathcal{O}$ for a class $\mathcal{C}$ of circuits is a worst-case VGB obfuscator for $\mathcal{C}$. In addition, we show that for classes $\mathcal{C}$ with some additional properties, $\mathcal{O}$ is in fact worst-case VBB. We refer the reader to Section 1.1 for an overview.

4.1 Definitions and Statement of Main Theorem

Notation and terminology. For a function $f : \{0,1\}^n \to \{0,1\}$, we say that a point $x \in \{0,1\}^n$ *separates* a circuit C from f if $C(x) \neq f(x)$. We say that a set $L \subseteq \{0,1\}^n$ separates C from f, if some $x \in L$ separates C from f. Given a circuit collection $\mathcal{K}$, we say that L separates $\mathcal{K}$ from f, if L separates any $C \in \mathcal{K}$ from f. Recall, that we say that a collection $\mathcal{K}$ is concentrated if the uniform distribution on $\mathcal{K}$ is concentrated around its majority function $\mathrm{maj}_{\mathcal{K}}$.

Definition 7 (Majority-separating oracle). *Let $\mathcal{C}$ be a collection of boolean circuits defined over $\{0,1\}^n$, let $C \in \mathcal{C}$, and let $\varepsilon > 0$. An oracle $\mathbb{S}$ is said to be $(\mathcal{C}, C, \varepsilon)$-separating if given any ε-concentrated sub-collection $\mathcal{K} \subseteq \mathcal{C}$, represented by a circuit that samples uniform elements in $\mathcal{K}$, $\mathbb{S}(\mathcal{K})$ outputs a point $x \in \{0,1\}^n$ that separates C from $\mathrm{maj}_{\mathcal{K}}$, or $\perp$ if no such point exists.*

Remark 2. In the above definition, and throughout this section, we often abuse notation and denote by $\mathcal{K}$ both the sub-collection and the circuit that samples uniform elements from the sub-collection.

Definition 8 (Learnability by majority-separating oracles). *A collection $\mathcal{C} = \bigcup_{n\in\mathbb{N}} \mathcal{C}_n$ of boolean circuits is said to be $(t, \boldsymbol{c}, s, \varepsilon)$-learnable by a majority-separation oracle if there exists a deterministic oracle-aided machine $\mathcal{L}$ such that, given oracle access to $C \in \mathcal{C}_n$ and a $(\mathcal{C}_n, C, \varepsilon(n))$-separating oracle $\mathbb{S}$, $\mathcal{L}^{C,\mathbb{S}}(1^n)$ outputs $\hat{C} \in \mathcal{C}_n$ of equivalent functionality to C, in time $t(n)$, using at most $s(n)$ queries to $\mathbb{S}$, and at most $\boldsymbol{c}_i(n)$ queries to C between the $i-1$-st and the i-th calls to $\mathbb{S}$.*

Our main technical theorem shows that any strong indistinguishability obfuscator for a circuit collection $\mathcal{C}$ that is learnable via a majority separation oracle is also a worst-case simulation-based obfuscator. The size and query complexity of the worst-case simulator, in particular whether it is a VBB or VGB simulator, is determined by the learnability parameters $(t, \boldsymbol{c}, s, \varepsilon)$.

Theorem 3. *Let $\mathcal{C} = \bigcup_{n\in\mathbb{N}} \mathcal{C}_n$ be a circuit collection that is $(t, \boldsymbol{c}, s, \frac{1}{q})$-learnable by a majority-separating oracle, for some polynomial q. Let $\mathcal{O}$ be a strong indistinguishability obfuscator for $\mathcal{C}$, let $\mathcal{A}$ be a boolean poly-size adversary, and let p be a polynomial. Then $(\mathcal{A}, p)$ has a simulator $\mathcal{S}$ of size $O(|\mathcal{A}| + t \cdot s \cdot q^s \cdot \prod_{i=1}^{s} 2^{\boldsymbol{c}_i})$ with $O(\|\boldsymbol{c}\|_1 + q \cdot s)$ oracle queries. The simulator works in the worst-case for any $C \in \mathcal{C}$.*

In Section 4.2 we show that any circuit collection $\mathcal{C}$ is indeed $(t, \boldsymbol{c}, s, \frac{1}{q})$-learnable, for some setting of parameters (where $\|\boldsymbol{c}\|_1, q, s$ are polynomially bounded).

4.2 VGB and VBB by Majority-Separation Learning

In this section, we show that any class of circuits is learnable by a majority-separating oracle, with parameters that yield VGB simulation. In the full version of this work we discuss additional classes that can be learned with better parameters, yielding VBB simulation. This includes previously obfuscated classes as well as new ones.

VGB Obfuscation for All Circuits. We show

Theorem 4. *Let $\mathcal{C}$ be any circuit collection and let $\mathcal{O}$ be a strong indistinguishability obfuscator for $\mathcal{C}$. Then $\mathcal{O}$ is also a worst-case VGB obfuscator for $\mathcal{C}$.*

To prove Theorem 4, we show that any circuit collection is learnable by a majority-separating oracle, where the learner is of unbounded size, but only performs a polynomial number of queries to its oracles. Theorem 4 then follows from Theorem 3.

Lemma 1. *For any $q > 2$, any circuit collection $\mathcal{C} = \{\mathcal{C}_n\}_{n\in\mathbb{N}}$ is $(t, \boldsymbol{c}, s, \frac{1}{q})$-learnable by a majority-separating oracle for $t(n) = \infty$, $s(n) \leq \|\boldsymbol{c}(n)\|_1 \leq q(n) \cdot \log|\mathcal{C}_n|$.*

5 From Semantically-Secure Graded Encodings to Strong IO for $\mathbf{NC^1}$

In this section we show that any semantically-secure graded encoding scheme, together with *any* ideal graded encoding obfuscation (i.e., any obfuscation that is virtual-black-box secure in the ideal encoding model) for a class $\mathcal{C}$ of circuits, implies *strong indistinguishability obfuscation* for $\mathcal{C}$.

Proposition 1. *Assume there exists a semantically-secure graded encoding scheme, and assume there exists an ideal graded encoding obfuscation for a circuit class $\mathcal{C}$. Then there exists a strong IO obfuscator for the circuit class $\mathcal{C}$, in the plain model (Definition 6).*

Our definition of semantically-secure graded encoding is a strengthening of the assumption of [PTS13]. One key difference from the assumption in [PTS13] is that here we consider semantic security even for distributions of messages that are not efficiently samplable. The definition can be found in the full version of this work where we also discuss several relaxations. The definition of VBB obfuscation in the ideal-graded-encoding model is also deferred to the full version of this work.

Proposition 1, combined with the recent VBB obfuscators for NC^1 in the ideal-graded-encoding model [BR13, BGK+13], and with our results from Section 4, immediately yields strong IO and VGB obfuscation for NC^1.

Theorem 5. *Assume there exists a semantically-secure graded encoding scheme. Then there exist strong IO and worst-case VGB obfuscation for any collection in NC^1.*

Acknowledgements. We are grateful to Rafael Pass for enlightening discussions and valuable comments.

References

[BBC+14] Barak, B., Bitansky, N., Canetti, R., Kalai, Y.T., Paneth, O., Sahai, A.: Obfuscation for evasive functions. In: Lindell, Y. (ed.) TCC 2014. LNCS, vol. 8349, pp. 26–51. Springer, Heidelberg (2014)

[BC10] Bitansky, N., Canetti, R.: On strong simulation and composable point obfuscation. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 520–537. Springer, Heidelberg (2010)

[BCC+14] Bitansky, N., Canetti, R., Cohn, H., Goldwasser, S., Kalai, Y.T., Paneth, O., Rosen, A.: The impossibility of obfuscation with auxiliary input or a universal simulator. CoRR, abs/1401.0348 (2014)

[BGI+01] Barak, B., Goldreich, O., Impagliazzo, R., Rudich, S., Sahai, A., Vadhan, S.P., Yang, K.: On the (im)possibility of obfuscating programs. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 1–18. Springer, Heidelberg (2001)

[BGK+13] Barak, B., Garg, S., Kalai, Y.T., Paneth, O., Sahai, A.: Protecting obfuscation against algebraic attacks. Cryptology ePrint Archive, Report 2013/631 (2013), http://eprint.iacr.org/

[BR13] Brakerski, Z., Rothblum, G.N.: Virtual black-box obfuscation for all circuits via generic graded encoding. Cryptology ePrint Archive, Report 2013/563 (2013), http://eprint.iacr.org/

[CLT13] Coron, J.-S., Lepoint, T., Tibouchi, M.: Practical multilinear maps over the integers. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 476–493. Springer, Heidelberg (2013)

[CV13] Canetti, R., Vaikuntanathan, V.: Obfuscating branching programs using black-box pseudo-free groups. Cryptology ePrint Archive, Report 2013/500 (2013), http://eprint.iacr.org/

[GGH13a] Garg, S., Gentry, C., Halevi, S.: Candidate multilinear maps from ideal lattices. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 1–17. Springer, Heidelberg (2013)

[GGH+13b] Garg, S., Gentry, C., Halevi, S., Raykova, M., Sahai, A., Waters, B.: Candidate indistinguishability obfuscation and functional encryption for all circuits. In: FOCS (2013)

[GK05] Goldwasser, S., Kalai, Y.T.: On the impossibility of obfuscation with auxiliary input. In: FOCS, pp. 553–562 (2005)

[GLSW14] Gentry, C., Lewko, A., Sahai, A., Waters, B.: Indistinguishability obfuscation from the multilinear subgroup elimination assumption. Cryptology ePrint Archive, Report 2014/309 (2014), http://eprint.iacr.org/

[Had00] Hada, S.: Zero-knowledge and code obfuscation. In: Okamoto, T. (ed.) ASIACRYPT 2000. LNCS, vol. 1976, pp. 443–457. Springer, Heidelberg (2000)

[PTS13] Pass, R., Telang, S., Seth, K.: Obfuscation from semantically-secure multilinear encodings. Cryptology ePrint Archive, Report 2013/781 (2013), http://eprint.iacr.org/

[SW13] Sahai, A., Waters, B.: How to use indistinguishability obfuscation: Deniable encryption, and more. IACR Cryptology ePrint Archive 2013, 454 (2013)

[Wee05] Wee, H.: On obfuscating point functions. IACR Cryptology ePrint Archive 2005, 1 (2005)