Refining Leakage Diagram Analyses for Random Probing Security
Extended version

Sonia Belaïd[1][0000−0002−9437−6425], Ghozlane Boukacem[1,2], Gaëtan Cassiers[1][0000−0001−5426−9345], Victor Normand[1,3][0009−0005−8862−9093], and Mélissa Rossi[1][0000−0002−9268−3034]

[1] CryptoExperts, Paris, France
[2] Sorbonne Université, CNRS, LIP6, F-75005 Paris, France
[3] DIENS, Ecole normale supérieure, PSL University, CNRS, Inria, Paris, France

Abstract. Masking is a widely used countermeasure against side-channel attacks, where sensitive variables are split into randomized shares in order to prevent information leakage from intermediate computations. Its security is commonly analyzed in the random probing model, in which each internal variable leaks independently with some probability p. Constructing large masked circuits that achieve random probing security (RPS) is challenging, and a common approach is to rely on composability frameworks ensuring that suitable local gadget-level properties imply circuit-level guarantees.
Recent works have introduced compositional frameworks based on leakage diagrams to derive security guarantees in the random probing model. While elegant and scalable, these approaches rely on conservative analyses and security proofs tailored to specific gadget constructions.
In this work, we revisit and refine a leakage-diagram approach published in TCHES 2023. We formalize and generalize the proof to show that the probability of an orbit in the leakage diagram upper bounds the adversary's RPS advantage. Our analysis identifies the local gadget-level conditions needed for this composition result, refines the analysis of edge inclusion probabilities in leakage diagrams for the underlying multiplication gadgets, leading to tighter final security bounds, and improves the combinatorial analysis of orbits through a detailed study of small cases and new structural properties for larger ones. We evaluate the resulting constructions on AES. Our results significantly improve the concrete security guarantees compared to previous leakage diagram analyses, while also improving the randomness complexity compared to existing composable approaches.

Keywords: Masking, Random Probing Security, Leakage Diagrams

1 Introduction

Side-Channel Leakage and Masking. In the standard cryptographic security model, adversaries are assumed to have access only to the inputs and outputs of an algorithm, without visibility into its internal computations. However, physical implementations leak information through side channels such as power consumption, electromagnetic emanations, or timing behavior. These side-channel attacks can compromise implementations even when the underlying primitives are secure. Masking [16,20] is a widely used countermeasure, which represents each sensitive variable as multiple randomized shares, forcing an attacker to combine information from multiple shares to recover any useful information. This countermeasure increases the measurement requirements exponentially in the number of shares [16].

Leakage Models and Random Probing. The security of masking schemes is analyzed through leakage models. The t-probing model [22] assumes that an adversary can observe a bounded number of internal variables, while the noisy leakage model [16,25] provides a more realistic but analytically challenging abstraction. The random probing model [22,18] offers a useful trade-off: each internal variable leaks independently with probability p. This model admits a reduction to the noisy leakage model and supports both simulation-based and quantitative analyses.

A central challenge is composability. Cryptographic circuits are built from smaller gadgets that can often be verified in isolation, but local security does not necessarily imply global security. Composability frameworks address this issue by ensuring that suitable local properties imply circuit-level guarantees.

An increasing number of works have focused on designing masking compilers achieving security in the random probing model [1,3,2,21,8,12,13,15,14,23,11,10]. However, only a few of these approaches provide concrete security guarantees for practical circuits of realistic size. A first line of work, initiated by [2] and later refined through the random probing expansion framework [8,12,13], achieves arbitrary security levels for constant leakage probabilities. These constructions rely on expansion-based techniques and provide asymptotic guarantees, but typically lead to large and costly circuits. More recently, Berti, Faust, and Orlt [14] developed a new approach based on *leakage diagrams*. Their compiler replaces each wire by an n-sharing, each gate by a corresponding gadget, and inserts circular refresh gadgets between every pair of gadgets. The resulting circuit is analyzed through a dependency graph whose subgraphs induced by the random probing leakage form the leakage diagrams. Their main result shows that the absence of orbits in the leakage diagram implies random probing security, leading to explicit analytical bounds as a function of the leakage probability p and the number of shares n. While elegant and scalable, this analysis remains relatively conservative. Alternative approaches have also been proposed. Jahandideh, Mennink, and Batina [23] introduced an algebraic framework based on mutual-information bounds. Their compiler, which also relies on circular refresh gadgets and variants of the multiplication gadget from [6], achieves improved security/performance trade-offs. However, its security is only heuristic, being based on empirically-extrapolated formulas. Recently, [24] proposes an analytical framework based on the leakage diagram of a succession of simple refreshes to reach tighter estimates in the random probing model. But a formal analysis of the multiplication gadget is left as future work and no formal compiler and bounds are available. Finally, an approach based on the direct verification has recently been introduced [7]. It can verify large circuits without relying on a composable framework.

Motivations. The leakage-diagram approach of [14] provides a powerful framework for analyzing masked circuits. However, its proof is tailored to specific gadgets and does not explicitly identify the underlying local security property required for composability. Moreover, its security bounds are conservative, relying on worst-case estimates that may over-approximate leakage propagation. This motivates a refined analysis that both isolates the required gadget-level property and improves the tightness of the resulting bounds.

Our Contributions. Our contributions can be summarized as follows:

- In Section 3, we formalize the masking compiler and the notions of dependency graph, leakage diagram, and orbit introduced in [14]. While doing so, we introduce the new notions of equivalent walks and arcs, which play a central role in our analysis.
- Then, in Section 4, we revisit and formalize the security proof of [14] to show that their compiler, instantiated with the circular refresh gadget together with affine and multiplication gadgets, achieves random probing security (RPS). More precisely, we prove that the adversary's advantage is upper bounded by the probability that the associated leakage diagram contains an orbit. In this framework, we identify and formalize the precise local security property required from non-refresh gadgets, which we call *output-robust simulatability* (ORS). We further introduce a simpler sufficient property, called *generalized free Strong Non Interference*, and prove that it implies ORS. Finally, we show that the multiplication gadget of [14] satisfies this property.
- In Section 5, we study the distribution of dependency edges in leakage diagrams under random probing leakage. We introduce a generic framework based on dependency mappings and dominating probabilities, together with a generic procedure for computing these probabilities. We instantiate this framework for affine gadgets and for the multiplication gadget from [14], based on a variant of the ISW multiplication of [22]. In particular, we refine the original manual analysis of [14] and derive tighter analytical security bounds for this construction.
- In Section 6, we improve the computation of upper bounds on the probability that a random probing leakage diagram associated with the masking compiler output contains an orbit. More precisely, we perform a detailed analysis of all possible orbits of size n, $n+1$, and $n+2$, and derive improved upper bounds for larger orbits of size at least $n+3$ using new structural properties of leakage diagrams.
- Finally in Section 7, we compare our approach with state-of-the-art random probing compilers and demonstrate significant improvements over both the compiler of [14] and the quantitative analysis of [23]. We also improve the randomness complexity compared to the compiler of [10].

2 Preliminaries

2.1 Notations

Throughout the paper, $\mathbb{K}$ denotes a finite field. For $n \in \mathbb{N}$, we write $[n] = \{0, \ldots, n-1\}$.

Sharings. We use the notation $[\![x]\!] = (x_0, \ldots, x_{n-1}) \in \mathbb{K}^n$ to denote a sharing of a secret value x. For a vector of secrets $\boldsymbol{x} = (x^1, \ldots, x^l)$, we write $[\![\boldsymbol{x}]\!] = ([\![x^1]\!], \ldots, [\![x^l]\!]) \in \mathbb{K}^{n \times l}$ to represent the collection of l sharings. For a subset of indices $I \subseteq [n]$, we denote by $[\![x]\!]_I = (x_i)_{i \in I}$ the restriction of the sharing $[\![x]\!]$ to the shares indexed by I. More generally, given a tuple of index sets $\boldsymbol{I} = (I^1, \ldots, I^l)$, we define $[\![\boldsymbol{x}]\!]_{\boldsymbol{I}} = \left([\![x^1]\!]_{I^1}, \ldots, [\![x^l]\!]_{I^l}\right)$ as the corresponding tuple of restricted sharings and, extending notation, for a single set I, $[\![\boldsymbol{x}]\!]_I = [\![\boldsymbol{x}]\!]_{(I,\ldots,I)}$.

Distributions. Two probability distributions D_1 and D_2 are ε-*close*, denoted $D_1 \approx_\varepsilon D_2$, if their statistical distance is at most ε, namely $\mathrm{SD}(D_1; D_2) \triangleq \frac{1}{2}\sum_x |p_{D_1}(x) - p_{D_2}(x)| \leq \varepsilon$, where $p_{D_1}(\cdot)$ and $p_{D_2}(\cdot)$ denote the probability mass functions of D_1 and D_2. We write $D_1 \overset{id}{=} D_2$ to denote that the two distributions are identical.

2.2 Masking and Security Notions

Linear Sharing, Circuits, and Gadgets. We introduce below the additive sharings and the arithmetic-circuit model underlying the masked gadgets considered throughout the paper.

Definition 1 (Additive Sharing). *Let $n, \ell \in \mathbb{N}$. For any $x \in \mathbb{K}$, an* n-sharing *of x is a vector $[\![x]\!] = (x_0, \ldots, x_{n-1}) \in \mathbb{K}^n$ such that $x_0 + \cdots + x_{n-1} = x$. It is said to be* uniform *if for any set $I \subseteq [n]$ with $|I| < n$, the tuple $[\![x]\!]|_I$ is uniformly distributed over $\mathbb{K}^{|I|}$. In that case, the distribution of $[\![x]\!]|_I$ is independent of x, so any strict subset of the n shares reveals no information about x. An* n-additive encoding *is a probabilistic algorithm* Enc *which, on input a tuple $\boldsymbol{x} = (x^1, \ldots, x^\ell) \in \mathbb{K}^\ell$, outputs a tuple $[\![\boldsymbol{x}]\!] = ([\![x^1]\!], \ldots, [\![x^\ell]\!]) \in (\mathbb{K}^n)^\ell$ such that $[\![x^i]\!]$ is a uniform n-sharing of x^i for every $i \in [\ell]$.*

An *arithmetic circuit* over a finite field (or ring) $\mathbb{K}$ is a labeled directed acyclic graph whose edges are *wires* and vertices are *arithmetic gates* operating on $\mathbb{K}$. The circuit is built from a set of gates $\mathbb{B} = \{g : \mathbb{K}^\ell \to \mathbb{K}^m\}$, including addition gates $(x^1, x^2) \mapsto x^1 + x^2$, linear gates $(x) \mapsto f(x)$ for linear functions f, multiplication gates $(x^1, x^2) \mapsto x^1 \cdot x^2$, and copy gates $x \mapsto (x, \ldots, x)$. A *randomized arithmetic circuit* additionally includes a random gate outputting a fresh uniform element of $\mathbb{K}$. For such a circuit C, the notation $C(\rho, \cdot)$ denotes the deterministic circuit obtained by fixing the random tape $\rho \in \{0,1\}^*$.

In the following, an (n-*share*, ℓ-*to-m*) *gadget* is a randomized arithmetic circuit mapping an input $[\![\boldsymbol{x}]\!] \in (\mathbb{K}^n)^\ell$ to an output $[\![\boldsymbol{y}]\!] \in (\mathbb{K}^n)^m$ such that

$$\boldsymbol{x} = \left(\sum_{k=0}^{n-1} x_k^1, \ldots, \sum_{k=0}^{n-1} x_k^\ell\right) \quad \text{and} \quad \boldsymbol{y} = \left(\sum_{k=0}^{n-1} y_k^1, \ldots, \sum_{k=0}^{n-1} y_k^m\right)$$

satisfy $\boldsymbol{y} = g(\boldsymbol{x})$ for some function g.

Random Probing Security. Let $p \in [0,1]$ denote the *leakage rate*. In the p-random probing model, the evaluation of an arithmetic circuit C leaks the value carried by each wire independently with probability p. Following [8], the random probing leakage of an arithmetic circuit is formally defined through two probabilistic algorithms.

- The *leaking-wires sampler* takes as input a circuit C and a leakage rate p, and outputs a set of wire labels $\mathcal{W} \leftarrow \mathsf{LeakingWires}(C, p)$, constructed by including each wire of C (including input and output wires) independently with probability p.

– The *assign-wires sampler* takes as input a circuit C, a set of wires $\mathcal{W}$, and a shared input $[\![\mathbf{x}]\!]$, and outputs $\mathbf{w} \leftarrow \mathsf{AssignWires}(C, \mathcal{W}, [\![\mathbf{x}]\!])$, where $\mathbf{w}$ contains the values carried by the wires in $\mathcal{W}$ during the randomized evaluation of C on input $[\![\mathbf{x}]\!]$.

We now recall the notions of *secret independence*, *random probing leakage* and *random probing security* from [8].

Definition 2 (Secret Independence). *A randomized arithmetic circuit C with ℓ n-share inputs is* secret independent *for a set of wires $\mathcal{W}$, with respect to a uniform additive encoding Enc, if there exists a simulator Sim such that, for every $\mathbf{x} \in \mathbb{K}^\ell$,*

$$\mathsf{Sim}(C, \mathcal{W}) \overset{id}{=} \mathsf{AssignWires}(C, \mathcal{W}, \mathsf{Enc}(\boldsymbol{x})).$$

Definition 3 (Random Probing Leakage). *Let C be a randomized arithmetic circuit and $[\![\mathbf{x}]\!] \in \mathbb{K}^{\ell \times n}$ a shared input. The* p-random probing leakage *of C on input $[\![\mathbf{x}]\!]$ is the distribution $\mathcal{L}_p(C, [\![\mathbf{x}]\!])$ defined by*

$$\mathcal{L}_p(C, [\![\mathbf{x}]\!]) \overset{id}{=} \mathsf{AssignWires}(C, \mathsf{LeakingWires}(C, p), [\![\mathbf{x}]\!]).$$

Definition 4 (Random Probing Security). *A randomized arithmetic circuit C with ℓ n-share inputs is* (p, ε)-random probing secure *with respect to a uniform additive encoding Enc if there exists a simulator Sim such that, for every $\mathbf{x} \in \mathbb{K}^\ell$,*

$$\mathsf{Sim}(C) \approx_\varepsilon \mathcal{L}_p(C, \mathsf{Enc}(\mathbf{x})).$$

Lemma 1 (Proof in Appendix A). *Let $p, \varepsilon \in [0, 1]$. Let C be a randomized arithmetic circuit and let Enc be a uniform additive encoding. Assume that*

$$\Pr_{\mathcal{W} \leftarrow \mathsf{LeakingWires}(C,p)}[C \text{ is secret independent for } \mathcal{W}] \geq 1 - \varepsilon.$$

Then C is (p, ε)-random probing secure with respect to Enc.

3 Formalization of Leakage Diagrams from [14]

In this section, we formalize the compiler, dependency graphs, leakage diagrams, and the related notions introduced in [14], which will serve as the foundation of our analysis. Based on these formalizations, we develop in Section 4 an alternative and more generic proof of the random probing security of the compiled circuits produced by these compilers.

Definition 5 (Compiler from [14]). *Let $n \in \mathbb{N}$ and let C be a (randomized) arithmetic circuit over $\mathbb{K}$ with gate set $\mathbb{B}$. We define the compiled circuit C_n as a randomized arithmetic circuit operating on n-sharings, obtained by the following transformations.*

– *Wire expansion. Each wire of C carrying a value $x \in \mathbb{K}$ is replaced by a bundle of n wires carrying an n-sharing $[\![x]\!] \in \mathbb{K}^n$.*
– *Affine gates. For any affine gate $g : \mathbb{K}^\ell \to \mathbb{K}^m$, we replace g by its* sharewise lifting *$\overrightarrow{g} : (\mathbb{K}^n)^\ell \to (\mathbb{K}^n)^m$. Let l_g be the linear map associated to g, for all input sharings $[\![\boldsymbol{x}]\!] \in \mathbb{K}^{\ell n}$ we define*

$$\overrightarrow{g}([\![\boldsymbol{x}]\!]) \triangleq (g(\boldsymbol{x}_0), l_g(\boldsymbol{x}_1), \ldots, l_g(\boldsymbol{x}_{n-1})).$$

This includes in particular linear gates, addition gates, and copy gates.
– *Multiplication gates. For any multiplication gate $g : \mathbb{K}^2 \to \mathbb{K}$ defined by $g(x, y) = x \cdot y$, we replace g by the n-share multiplication gadget of [14], which, on input $([\![x]\!], [\![y]\!]) \in (\mathbb{K}^n)^2$, outputs an n-sharing $[\![z]\!]$ such that $\sum_{i=0}^{n-1} z_i = \sum_{i=0}^{n-1} x_i \cdot \sum_{i=0}^{n-1} y_i$.*

Algorithm 1 Circular refresh [5]

Input: $[\![x]\!] = (x_0, \ldots, x_{n-1})$.
Output: $[\![y]\!] = (y_0, \ldots, y_{n-1})$ such that $\sum_{i=0}^{n-1} y_i = \sum_{i=0}^{n-1} x_i$.
  for $i = 0$ to $n-1$ **do**
    sample $r_i \xleftarrow{\$} \mathbb{K}$
  for $i = 0$ to $n-1$ **do**
    $b_i \leftarrow x_i + r_i$
    $y_i \leftarrow b_i - r_{i-1 \bmod n}$
  return $[\![y]\!]$

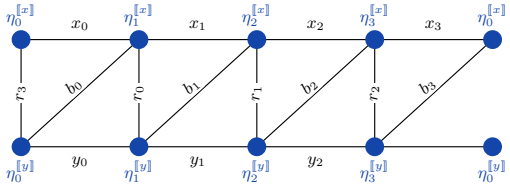


Fig. 1: Circular refresh gadget [5] and its dependency graph for $n = 4$. The graph is flattened to a line for readability, but the leftmost and rightmost vertices are identified.

– *Refresh insertion. For every n-sharing connecting the output of a gadget to the input of another gadget, we insert a circular refresh gadget (from [5], recalled in Figure 1 where the indices are considered modulo n) acting on the corresponding n-sharing.*

The resulting circuit C_n is a composition of affine and multiplication gadgets interconnected exclusively through circular refresh gadgets. We denote by $\mathcal{G}_\alpha^{C_n}$ the set of gadgets of type α in C_n, where $\alpha \in \{\mathrm{lin}, \mathrm{add}, \mathrm{mult}, \mathrm{copy}\}$, with $\mathcal{G}_{\mathrm{lin}}$ denoting linear 1-to-1 gadgets, $\mathcal{G}_{\mathrm{add}}$ denoting 2-to-1 addition gadgets, $\mathcal{G}_{\mathrm{mult}}$ denoting multiplication gadgets, and $\mathcal{G}_{\mathrm{copy}}$ denoting copy gadgets.

Definition 6 (Dependency graph). *Let C be an arithmetic circuit and let C_n be its compiled version as in Definition 5. The* dependency graph *of C, denoted Γ_{C_n}, is the undirected graph defined as follows.*

– *Non-refresh gadgets. For each non-refresh gadget G in C_n, we introduce n vertices $\eta_0^G, \ldots, \eta_{n-1}^G$, and n edges $\nu_0^G, \ldots, \nu_{n-1}^G$, where, for all $i \in [n]$ (indices taken modulo n), the edge ν_i^G connects the vertices η_i^G and η_{i+1}^G. Each edge ν_i^G represents the i-th share.*
– *Refresh edges. For each refresh gadget R connecting a gadget G to a gadget H, we add edges between the corresponding vertices according to the dependency structure of the circular refresh (see Figure 1). Concretely, for all $i \in [n]$, we add the edges $\{\eta_i^G, \eta_i^H\}$ (labelled by r_{i-1}^R) and $\{\eta_i^G, \eta_{i+1}^H\}$ (labelled by b_i^R).*
– *Composition. The graph Γ_{C_n} is obtained as the union of all gadget graphs and all refresh edges.*

Figure 1 illustrates a circular refresh gadget and its dependency graph for $n = 4$. Figure 2 illustrates the dependency graph of a simple compiled circuit C_n composed of six gadgets $(G_i)_{1 \le i \le 6}$ connected through circular refresh gadgets, where G_5 is a two-input gadget (*e.g.*, an addition or multiplication gadget). For readability, all vertices are labeled, whereas only the edges of G_1, G_3, and of their intermediate refresh gadget R_1 are explicitly labeled. In particular, the r- and b-edges should be understood with superscript R_1.
In the following definitions, we consider $n \in \mathbb{N}$, an arithmetic circuit C, its compiled circuit C_n, and the associated dependency graph Γ_{C_n}.

Definition 7 (Leakage diagram). *Let $\mathcal{W}$ be a set of wires of C_n. Let Φ[4] be a mapping assigning to any subset of wires of a multiplication gadget a subset of indices in $[0, n-1]$. The* $(\mathcal{W}, \Phi, C_n)$-leakage diagram *is the subgraph $\Gamma_{C_n,\Phi}^{\mathcal{W}} \subseteq \Gamma_{C_n}$ defined as follows.*

– *Refresh gadgets. Since all internal wires of refresh gadgets are explicitly represented in Γ_{C_n}, the leakage diagram $\Gamma_{C_n,\Phi}^{\mathcal{W}}$ contains exactly the edges corresponding to the wires in $\mathcal{W}$ belonging to refresh gadgets.*
– *Affine gadgets. Let G be an affine gadget and let $\mathcal{W}_G \subseteq \mathcal{W}$ be the subset of leaking wires of G. For every wire in $\mathcal{W}_G$ corresponding to the part of the gadget with share index $i \in [n]$, the leakage diagram $\Gamma_{C_n,\Phi}^{\mathcal{W}}$ contains the edge ν_i^G.*

[4] For simplicity, we assume that all multiplication gadgets are identical. The definition naturally extends to heterogeneous multiplication gadgets by considering one mapping per gadget type.

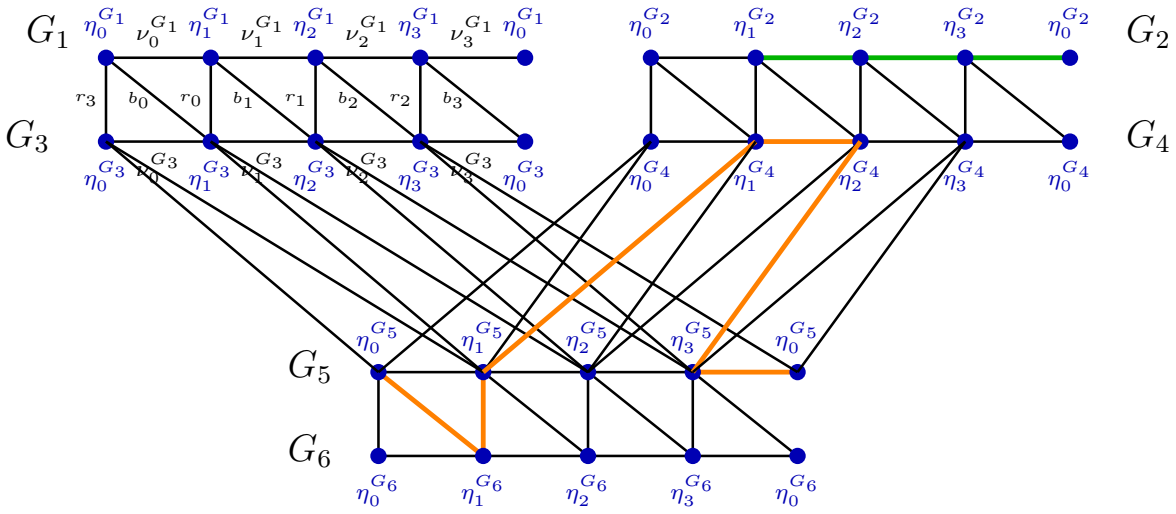


Fig. 2: Illustration of a dependency graph Γ_{C_n} (Definition 6).

– *Multiplication gadgets. Let G be a multiplication gadget and let $\mathcal{W}_G \subseteq \mathcal{W}$ be the subset of leaking wires of G. The leakage diagram $\Gamma^{\mathcal{W}}_{C_n,\Phi}$ contains all edges $\{\nu_i^G \mid i \in \Phi(\mathcal{W}_G)\}$.*

Definition 8 (Equivalent Walks). *We consider non-empty finite walks in the dependency graph Γ_{C_n}, defined by a sequence of consecutive edges: $\omega = (e_0, \ldots, e_{k-1})$, with associated vertices $(v_0, \ldots, v_k)$. If the last vertex of a walk $\omega = (e_0, \ldots, e_{k-1})$ coincides with the first vertex of a walk $\omega' = (e'_0, \ldots, e'_{k'-1})$, the two walks can be concatenated, and their concatenation, denoted $\omega \| \omega'$, is $(e_0, \ldots, e_{k-1}, e'_0, \ldots, e'_{k'-1})$. The reversal of a walk $\omega = (e_0, \ldots, e_{k-1})$, denoted $-\omega$ is $(e_{k-1}, \ldots, e_0)$.*

Walk equivalence, denoted $\sim$, is an equivalence relation between walks that:

– *is preserved by concatenation and reversal: if $\omega_1 \sim \omega'_1$ and $\omega_2 \sim \omega'_2$, then $\omega_1 \| \omega_2 \sim \omega'_1 \| \omega'_2$ and $-\omega_1 \sim -\omega'_1$,*
– *allows cancellation: for any edges e and e', $(e, e', e') \sim (e)$, and*
– *contains sum-equivalent paths in refresh gadgets: for every refresh gadget R connecting a gadget G to a gadget H and every $i \in [n]$, $(\nu_i^G, r_i^R) \sim (b_i^R)$ and $(b_i^R) \sim (r_{i-1 \mod n}^R, \nu_i^H)$.*

Definition 9 (Arc). *Let G be a gadget in C_n. For any $i, j \in \mathbb{Z}$ with $i \leq j$, the arc $\alpha_{i,j}^G$ over the circle[5] of G is the walk $\alpha_{i,j}^G \triangleq (\nu_k^G)_{k=i,\ldots,j}$.*

Definition 10 (Orbit). *An* orbit *is a walk in Γ_{C_n} which is equivalent to an arc $\alpha_{i,i+kn-1}^G$ for a gadget G in C_n and some i and k with $k > 0$.*

For instance, in Figure 2, the green highlighted path corresponds to the arc $\alpha_{1,3}^{G_2}$ and the orange highlighted path corresponds to an orbit.

4 New Simulation-Based Proof

In this section, we revisit the proof of [14] and provide an alternative simulation-based proof of the random probing security of the compiled circuits (Definition 5). The structure of the proof is summarized in Figure 3. The full proofs for the claims in this section are given in Appendix B.

[5] By circle, we mean the set of horizontal edges. Even if they are drawn as a line for readability in this paper, they form a closed loop for all gadgets.

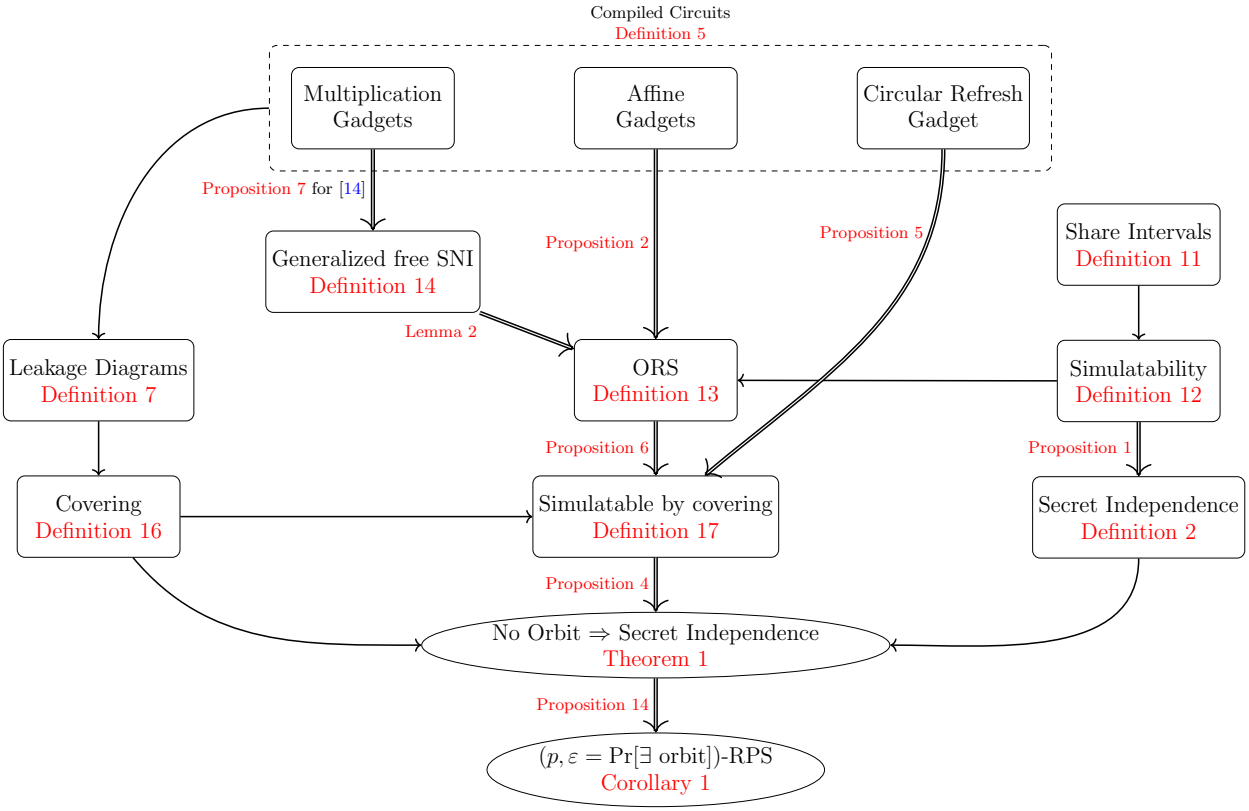


Fig. 3: Overview of the security proof. Double arrows indicate proved implications, while single arrows indicate necessary notions. The dashed box groups the components of the compiled circuit.

The proof starts from the local building blocks of the compiler, namely affine gadgets, multiplication gadgets, and circular refresh gadgets. In Section 4.1, we first introduce a general notion of simulatability for a gadget or a circuit, expressing which sums of subsets of input shares are sufficient to simulate a given set of wires and sums of subsets of output shares. This notion relies on share intervals and partial sums (*i.e.*, the sum of the shares whose indices belong to an interval modulo n), which naturally capture the structure induced by the circular refresh gadgets and the leakage diagrams. It can also be viewed as a generalization of the classic simulation that underlies many security notions in probing models, such as (strong-) non-interference [4] or random probing composability [8].

Building on this framework, we define the notion of output-robust simulatability (ORS) satisfied by affine gadgets and identify a stronger, yet easier to verify, property for multiplication gadgets, which we call *generalized free strong non-interference (generalized free-SNI)*. This makes the proof modular: our arguments are not tied to the specific multiplication gadget of [14], but apply to any multiplication gadget satisfying this property. Using standard sequential and parallel composition arguments, we then show that simulatability propagates through the whole compiled circuit.

In Section 4.2, we turn to the graph-theoretic side of the proof and introduce coverings of leakage diagrams. In a nutshell, a covering records the connectivity between the nodes of the input sharings, given some connections between the output nodes. The gadgets are covering simulatable (Section 4.3), which intuitively means that the covering corresponds to the set of share intervals whose partial sums are sufficient to simulate all probes represented in a leakage diagram along with some partial sums of the output sharings. Since coverings of gadget compositions behave similarly to simulatability, it follows that a composition of covering simulatable gadgets is covering simulatable. Overall, Section 4.3 establishes that the presence of an orbit

in the leakage diagram corresponds to the existence of a complete share interval in the covering, which is precisely the situation where the simulated values may depend on the underlying secret.

Finally, in Section 4.4, we move to the random probing setting, where the leakage diagram becomes a random object. The probability that it contains an orbit then bounds the probability of simulation failure, yielding the final (p, ε)-random probing security statement (Corollary 1), where ε upper bounds the probability of an orbit at leakage rate p.

4.1 Share Interval Simulation

We first introduce a notion of *simulatability* that captures how internal probes and output shares of a gadget can be simulated from a restricted subset of its input shares. This notion will serve as the main tool in our compositional analysis and will allow us to express security in the random probing model in a modular way. For linking random probing security and orbits in a leakage diagram, we will need to consider partial sums of shares, that will correspond to arcs in the leakage diagram. We thus introduce the notion of share intervals, which are sets of consecutive shares (indices taken modulo n) that can be added together to get partial sums.

Definition 11 (Share interval). *Given a number of shares n, we consider the equivalence relation $\sim$: for any $i_1, i_2, j_1, j_2 \in \mathbb{Z}$, $[i_1, j_1] \sim_n [i_2, j_2]$ if and only if there exists $k \in \mathbb{Z}$ such that $i_2 = i_1 + kn$ $j_2 = j_1 + kn$. A* share interval *is an equivalence class of $\sim_n$. For $i \leq j$, we denote by (i, j) the share interval in which $[i, j]$ lies, and any element of the class is named an* instance *of the share interval. We represent share intervals by any of their instances. A* complete *share interval is an interval of the form $(i, i + kn - 1)$ for some $k > 0$. Let us denote by Ω the set of complete share intervals.*

Given two share intervals, their sum is defined if they admit a representation (i_1, j_1) and (i_2, j_2) such that $i_2 = j_1 + 1$, it is denoted $(i_1, j_1) \oplus (i_2, j_2)$ and is equal to (i_1, j_2). Similarly the difference $(i_1, j_1) \ominus (i_2, j_2)$ is defined as $(j_2 + 1, j_1)$ when $i_1 = i_2$ and $j_2 < j_1$, and defined as $(i_1, i_2 - 1)$ when $j_1 = j_2$ and $i_1 < i_2$.

We say that a share interval (i, j) is a *singleton* if $j = i$.

For example, if $n = 5$, we have a singleton share interval $(0, 0)$, a share interval of size 2 $(1, 2)$, etc. The share intervals $(0, 4)$, $(1, 5)$, or even $(0, 9)$ are complete and therefore belong to the infinite set Ω. By contrast the share interval $(0, 5)$ is not complete. The sum $(3, 5) \oplus (1, 2)$ is defined and equal to $(1, 5)$, while the difference $(1, 4) \ominus (3, 4)$ is defined and equal to $(1, 2)$. The difference $(6, 9) \ominus (3, 4)$ is the exact same operation, using a different notation for the first operand.

Partial sum of a share interval. The *partial sum* of a sharing $[\![x]\!]$ for a share interval (i, j) is

$$[\![x]\!]_{(i,j)} \triangleq \sum_{k=i}^{j} x_{k \bmod n},$$

It is well-defined as it is identical for all instances of a share interval. By extension, for an (ordered) set of share intervals S, we denote $[\![x]\!]_S = \{[\![x]\!]_s : s \in S\}$.[6]

Closure of sets of share intervals. Further, the closure $\overline{S}$ of a set of share intervals S is the minimal set such that $S \subset \overline{S}$, and for any $x, y \in \overline{S}$ such that $x \oplus y$ (respectively $x \ominus y$) is defined, $x \oplus y \in \overline{S}$ (resp. $x \ominus y \in \overline{S}$). The closure of some sets is illustrated in Table 1.

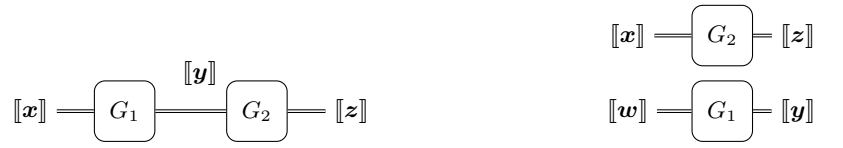
Let us consider another example with $S = \{(0, 3), (0, 1), (4, 1)\}$. The closure $\overline{S} = \overline{\{(0, 1), (2, 3), (4, 4)\}}$. It contains the complete share interval $(0, 4)$ and actually $\Omega \subset \overline{S}$.

Remark 1. The addition and subtraction of share intervals correspond to the addition and subtraction of the corresponding partial sums. Therefore, the partial sums of the intervals in the closure of a set S correspond to all partial sums that can be deduced from the partial sums of the intervals in S. From a probing-security point of view, the closure $\overline{S}$ does not reveal more information than S.

[6] Let us remark that looking at partial sums on share intervals specifically comes from the refresh gadget we are using, and the induced structure of the leakage diagrams. Although not necessary in this work, our security notions generalize to arbitrary multisets of share indices.

Table 1: Examples of share interval set closure for $n = 5$.

S	$\overline{S}$
$\{(4,0),(1,1),(3,3)\}$	$\{(4,0),(1,1),(3,3),(4,1),(3,0),(3,1)\}$
$\{(0,3),(0,1)\}$	$\{(0,3),(0,1),(2,3)\}$



(a) Sequential composition $G_2 \circ G_1$.

(b) Parallel composition $G_1 \| G_2$.

Fig. 4: Compositions of the gadgets G_1 and G_2.

Definition 12 (Simulatability). *Let $n \in \mathbb{N}$ and let G be a gadget mapping input n-sharings $[\![\boldsymbol{x}]\!] = ([\![x^0]\!], \ldots, [\![x^{l-1}]\!]) \in \mathbb{K}^{nl}$ to output n-sharings $[\![\boldsymbol{y}]\!] = ([\![y^0]\!], \ldots, [\![y^{m-1}]\!]) \in \mathbb{K}^{nm}$. Let $\boldsymbol{I} = (I^0, \ldots, I^{l-1})$ and $\boldsymbol{O} = (O^0, \ldots, O^{m-1})$, where I^i and O^j are sets of share intervals for each input and output sharing, and let $\mathcal{W}$ be a set of internal wires of G. We say that G is $(\boldsymbol{I}, \boldsymbol{O}, \mathcal{W})$-simulatable if there exists a simulator Sim such that, for all input sharings $[\![\boldsymbol{x}]\!]$,*

$$\mathsf{Sim}([\![\boldsymbol{x}]\!]_{\boldsymbol{I}}) \overset{id}{=} \mathsf{AssignWires}(G, \mathcal{W} \cup [\![\boldsymbol{y}]\!]_{\boldsymbol{O}}, [\![\boldsymbol{x}]\!]),$$

where $[\![\boldsymbol{x}]\!]_{\boldsymbol{I}}$ (resp. $[\![\boldsymbol{y}]\!]_{\boldsymbol{O}}$) denotes the collection of partial sums corresponding to the intervals in $\boldsymbol{I}$ (resp. $\boldsymbol{O}$). In other words, the joint distribution of the probed internal wires and the output partial sums $[\![\boldsymbol{y}]\!]_{\boldsymbol{O}}$ can be simulated given only the input partial sums $[\![\boldsymbol{x}]\!]_{\boldsymbol{I}}$.

In practice, the set of share intervals $\boldsymbol{I}$ depends on the set of wires $\mathcal{W}$ that we want to simulate, and we will often write $\boldsymbol{I}(\mathcal{W})$ to make this dependence explicit.

This simulatability notion is a generalization of the classic simulatability at the core of many composable notions in probing models that uses sets of shares, which amounts to using exclusively singleton share intervals in our definition. For example, in the t-probing model [4], a 1-to-1 gadget G is NI (resp. SNI) if and only if for any $\mathcal{W}$ and O such that $|\mathcal{W}| + |O| \leq t$, there exists I, defined here as a singleton union such that $|I| \leq t$ (resp. $|I| \leq |\mathcal{W}|$) and G is $(I, O, \mathcal{W})$-simulatable. Similarly, a 1-to-1 gadget G is (ε, t, p)-RPC if and only if for any O defined as a singleton union such that $|O| \leq t$, there exists a simulator that given $\mathcal{W} \leftarrow \mathsf{LeakingWires}(G, p)$ outputs a singleton union I such that $Pr(|I| \leq |\mathcal{W}|) \leq \varepsilon$ and G is $(I, O, \mathcal{W})$-simulatable. We refer to [8] for the full definition of RPC.

We are now able to derive composability properties, illustrated in Figure 4, for a composition of gadgets G_1 and G_2. Let $\mathcal{W}_1$ and $\mathcal{W}_2$ be sets of internal wires in G_1 and G_2, respectively. For the sequential composition $G_2 \circ G_1$, if G_1 is $(\boldsymbol{I_1}, \boldsymbol{O_1}, \mathcal{W}_1)$-simulatable and G_2 is $(\boldsymbol{O_1}, \boldsymbol{O_2}, \mathcal{W}_2)$-simulatable, then the composition is $(\boldsymbol{I_1}, \boldsymbol{O_2}, \mathcal{W}_1 \cup \mathcal{W}_2)$-simulatable. For the parallel composition $G_1 \| G_2$, G_1 is $(\boldsymbol{I_1}, \boldsymbol{O_1}, \mathcal{W}_1)$-simulatable and G_2 is $(\boldsymbol{I_2}, \boldsymbol{O_2}, \mathcal{W}_2)$-simulatable, then the composition is $((\boldsymbol{I_1}, \boldsymbol{I_2}), (\boldsymbol{O_1}, \boldsymbol{O_2}), \mathcal{W}_1 \cup \mathcal{W}_2)$-simulatable. The formal statements for these composition properties and their proofs are detailed in Appendix B.

Next, we give a condition under which simulatability implies secret independence, i.e., when it guarantees security.

Proposition 1 (Secret Independence from Simulatability). *Let $n \in \mathbb{N}$, let C be a ℓ_1-to-ℓ_2 gadget and $\mathcal{W}$ be a set of wires of C. Let $I_i(\mathcal{W})$ be such that C is $((I_1(\mathcal{W}), \ldots, I_{\ell_1}(\mathcal{W})), \emptyset, \mathcal{W})$-simulatable (see Definition 12). Then C is secret independent (as defined in Definition 2) for $\mathcal{W}$ and with respect to a uniform additive encoding if for all $i \in \{1, \ldots, \ell_1\}$, $\overline{I}_i(\mathcal{W}) \cap \Omega = \emptyset$, i.e., the closure of $I_i(\mathcal{W})$ does not contain any complete interval.*

Proof (sketch). We can show that every $[\![x]\!]^i_{I_i(\mathcal{W})}$ is independent of x^i: since $\overline{I}_i(\mathcal{W}) \cap \Omega = \emptyset$, one cannot recover x^i from a linear combination of the partial sums. □

Then, we introduce the notion of *output-robust simulatability* which ensures that the set of input shares required to simulate internal probes is stable with respect to additional output probes. Intuitively, probing an output share only requires revealing the corresponding input share with the same index.

Definition 13 (Output-Robust Simulatability (ORS)). *Let $n \in \mathbb{N}$ and let G be a (n-share, ℓ-to-m) gadget. Let I be a set of share intervals, and let $\mathcal{W}$ be a set of internal wires. We say that G is output-robust $(I, \mathcal{W})$-simulatable if, for every set of share intervals O, the gadget G is*

$$\big((I \cup O)^\ell,\ O^m,\ \mathcal{W}\big)\text{-simulatable},$$

where $(I \cup O)^\ell$ (resp. O^m) denotes the ℓ-tuple (resp. m-tuple) obtained by repeating $I \cup O$ (resp. O).

First, all affine gadgets are ORS.

Proposition 2 (ORS of Affine Gadgets). *Let $n \in \mathbb{N}$. Let G be a gadget mapping the input n-sharings $[\![\boldsymbol{x}]\!]$ to output n-sharings $[\![\boldsymbol{y}]\!]$, which is the compilation of an affine gate (Definition 5). Then, for any set of wires $\mathcal{W}$ in G, there exists a set of input indices $I \subseteq [n]$ such that $|I| \leq |\mathcal{W}|$ and G is output-robust $(\{(i,i) : i \in I\}, \mathcal{W})$-simulatable.*

Proof (sketch). Simulation of the wires in $\mathcal{W}$ directly follows from the sharewise structure of the gadget, and the simulation of an output partial sum from the corresponding input partial sum additionally uses the affine property of each subcircuit. □

Next, for the multiplication gadgets, ORS will follow from a variant of free-SNI [17] that has a relaxed relation between the number of probes and the set of share indices.

Definition 14 (Generalized Free SNI). *Let $n \in \mathbb{N}$, let G be a (n-share, ℓ-to-m) gadget mapping input n-sharings $[\![\boldsymbol{x}]\!]$ to output n-sharings $[\![\boldsymbol{y}]\!]$. Let $\mathcal{W}$ be a set of wires of G, and let $I \subseteq [n]$. G is $(I, \mathcal{W})$-generalized free-SNI if*

(i) there exists a simulator Sim such that for all input sharings $[\![\boldsymbol{x}]\!]$ and corresponding output sharing $[\![\boldsymbol{y}]\!]$,

$$\mathsf{Sim}\left([\![\boldsymbol{x}]\!]_I\right) \overset{id}{=} \mathsf{AssignWires}\left(G, \mathcal{W} \cup [\![\boldsymbol{y}]\!]_I, [\![\boldsymbol{x}]\!]\right)$$

(ii) for all $[\![\boldsymbol{x}]\!]$, for all $O \subsetneq [n] \backslash I$, $[\![\boldsymbol{y}]\!]_O$ is uniformly and independently distributed, conditioned on $\mathsf{AssignWires}(G, \mathcal{W} \cup [\![\boldsymbol{y}]\!]_I, [\![\boldsymbol{x}]\!])$.

Lemma 2 (ORS from Generalized Free SNI). *For any set of wires $\mathcal{W}$ in a gadget G, if I is such that G is $(I, \mathcal{W})$-generalized free-SNI, then G is output-robust $(I', \mathcal{W})$-simulatable, where $I' = \{(i,i) : i \in I\}$.*

Proof (sketch). The generalized free-SNI simulator can be turned into an ORS simulator: we can exploit the joint uniformity of the output shares to simulate any set of output partial sums whose closure (when joined with I') is not complete. □

4.2 Leakage Diagrams Covering

Let us now turn back to the leakage diagrams. We formalize the notion of *covering*, which will allow us to relate the notion of simulation to properties of the leakage diagrams. At a high level, a covering models the set of all possible walks in a leakage diagram that connect vertices on the input sharings, with an exception for the case where the diagram contains an orbit (regardless whether this orbit is connected to the input). Intuitively, the covering of a circuit consists of sets of share intervals on its input sharings, corresponding to

the partial sums required to simulate the leakage, together with partial sums on the output sharings. This correspondence will be established in Section 4.3.

In this section, we fix $n \in \mathbb{N}$, an arithmetic circuit C, its compiled circuit C_n, the associated dependency graph Γ_{C_n}, a mapping Φ associated with the multiplication gadgets of C_n such that for every multiplication gadget G in C_n and every leakage set $\mathcal{W}_G$ of wires of G, the gadget G is $(\Phi(\mathcal{W}_G), \mathcal{W}_G)$-generalized free-SNI. Let $\mathcal{W}$ be a set of wires of C_n. We denote by $\Gamma^{\mathcal{W}}_{C_n,\Phi}$ the corresponding $(\mathcal{W}, \Phi, C)$-leakage diagram.

In order to discuss the composition properties, we need to consider partial sums of share intervals on output sharings. For this purpose, we extend the definition of leakage diagrams accordingly. Intuitively, we add some "shortcuts" to the diagram that represent the given partial sums of the output sharings. An example of an extended leakage diagram is given in Figure 8 in Appendix B.

Definition 15 (Extended Leakage Diagram). *Let $[\![\boldsymbol{y}]\!] = ([\![y^0]\!], \ldots, [\![y^{m-1}]\!])$ be the output sharings of C_n and let $\boldsymbol{O} = (O^0, \ldots, O^{m-1})$ be a tuple of sets of share intervals. The extended leakage diagram of C_n with respect to $\mathcal{W}$, Φ and $\boldsymbol{O}$, denoted $\Gamma^{\mathcal{W},\boldsymbol{O}}_{C_n,\Phi}$, is a graph obtained by adding edges to $\Gamma^{\mathcal{W}}_{C_n,\Phi}$ that correspond to the output partial sums. More precisely, an extra edge denoted $\nu^{[\![y^i]\!]}_{(a,b)}$ is added between $\eta^{[\![y^i]\!]}_{a}$ and $\eta^{[\![y^i]\!]}_{b+1 \bmod n}$ for each $(a,b) \in O^i$, for all i. A walk along that edge is considered equivalent to the corresponding arc: $\nu^{[\![y^i]\!]}_{(a,b)} \sim \alpha^{[\![y^i]\!]}_{(a,b)}$.*

Definition 16 (Covering). *Let $[\![\boldsymbol{x}]\!] = ([\![x^0]\!], \ldots, [\![x^{\ell-1}]\!])$ be the input sharings of C_n, and let $\boldsymbol{O} = (O^0, \ldots, O^{m-1})$ be a tuple of sets of share intervals.*

For every input sharing index $i \in [\ell]$, the covering *of the extended leakage diagram $\Gamma^{\mathcal{W},\boldsymbol{O}}_{C_n,\Phi}$ on the input sharing $[\![x^i]\!]$, denoted $\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}$, is the set of share intervals defined as follows[7]:*

- *If the extended leakage diagram $\Gamma^{\mathcal{W},\boldsymbol{O}}_{C_n,\Phi}$ contains an orbit, then $\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i} = \Omega^*$ for all i, where $\Omega^* \triangleq \{(j, j+k) : j, k \in \mathbb{N}\}$.*
- *Otherwise, the covering $\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}$ is the set of all share intervals (a,b) for which there exists a walk in the extended leakage diagram $\Gamma^{\mathcal{W},\boldsymbol{O}}_{C_n,\Phi}$ that is equivalent to an arc $\alpha^{[\![x^i]\!]}_{a,b}$ on the input circle of $[\![x^i]\!]$.*

We denote the tuple of all coverings of a gadget: $\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n} \triangleq \left(\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}\right)_i$.

Let us first remark that a covering is always closed, i.e., for any covering $\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}$, $\overline{\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}} = \mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}$ (see proof in Appendix B).

Moreover, the coverings follow naturally composition laws under gadget composition, as proven in the same Appendix: for $C = C_2 \circ C_1$, $\mathcal{C}^{\mathcal{W}_1 \cup \mathcal{W}_2,\boldsymbol{O}}_{C} \supset \mathcal{C}^{\mathcal{W}_1,\boldsymbol{I_2}}_{C_1}$ where $\boldsymbol{I_2} = \mathcal{C}^{\mathcal{W}_2,\boldsymbol{O}}_{C_2}$, and for $C = C_1 \| C_2$, $\mathcal{C}^{\mathcal{W}_1 \cup \mathcal{W}_2,(\boldsymbol{O},\boldsymbol{O}')}_{C_1 \| C_2} \supset \left(\mathcal{C}^{\mathcal{W}_1,\boldsymbol{O}}_{C_1}, \mathcal{C}^{\mathcal{W}_2,\boldsymbol{O}'}_{C_2}\right)$, where the set inclusion is element-wise on the tuples.

We can now relate orbits and complete coverings, which is the first link in the chain that relates orbits to random probing security.

Proposition 3 (Equivalence between Orbits and Complete Coverings). *With the above notations, the following statements are equivalent:*

1. *the leakage diagram $\Gamma^{\mathcal{W}}_{C_n,\Phi}$ (or equivalently $\Gamma^{\mathcal{W},(\emptyset,\ldots,\emptyset)}_{C_n,\Phi}$) contains an orbit;*
2. *there exists an input sharing index $i^\star$ such that $\mathcal{C}^{\mathcal{W}}_{C_n,i^\star} \cap \Omega \neq \emptyset$.*

Proof. The implication from left to right is immediate from the definition of the covering. Conversely, assume that there exists an input sharing index $i^\star$ such that $\mathcal{C}^{\mathcal{W}}_{C_n,i^\star} \cap \Omega \neq \emptyset$. Then there exists a complete share interval $(j^\star, j^\star + kn - 1) \in \mathcal{C}^{\mathcal{W}}_{C_n,i^\star}$ for some $j^\star, k > 0$. By definition of the covering, this means that either the leakage diagram contains an orbit, or it contains a walk equivalent to the arc $\alpha^{[\![x^{i^\star}]\!]}_{j^\star, j^\star+kn-1}$, meaning that it is an orbit. □

[7] For readability, we omit Φ from the notation.

4.3 Secret Independence from Absence of Orbits

We now connect the simulatability notions with the diagrams definitions, and in particular with their coverings.

Throughout this section, coverings are defined with respect to the $(\mathcal{W}, \Phi, C_n)$-leakage diagrams of compiled circuits C_n. For readability, we omit the dependency on Φ from the notation.

Definition 17 (Covering Simulatability). *Let C be an ℓ-to-m gadget. We say that C is* simulatable by its covering *(or equivalently* covering-simulatable*) if, for every leakage set $\mathcal{W}$ and every m-tuple of sets of share intervals $\boldsymbol{O} = (O^0, \ldots, O^{m-1})$, the gadget C is $(\mathcal{C}_C^{\mathcal{W},\boldsymbol{O}}, \boldsymbol{O}, \mathcal{W})$-simulatable.*

Since simulatability and coverings follows similar composition properties, we can prove (in Appendix B) that any composition of covering-simulatable gadgets is itself covering-simulatable.

Proposition 4 (Covering Simulatability for composition). *Any arbitrary composition of covering-simulatable gadgets is covering-simulatable.*

We next prove that all our gadgets are covering simulatable. For refresh gadgets, we explicitly build the simulation in Appendix B.3, while for the non-refresh gadgets, this follows from ORS.

Proposition 5 (Covering Simulatability of the Circular Refresh Gadget). *The circular refresh gadget, recalled in Figure 1, is simulatable by its covering.*

Proposition 6 (Covering Simulatability of (non-refresh) ORS Gadgets). *Let G be a gadget, $\mathcal{W}$ a set of wires in G, $I \subset [n]$ and $I' = \{(i, i) : i \in I\}$ such that G is $(I', \mathcal{W})$-output-robust simulatable. If the leakage diagram for G is the subset of edges of a circle whose indices belong to I, then G is simulatable by its covering.*

Proof (sketch). The covering contains I' and the set of output share intervals to simulate, which is exactly the input needed by the ORS simulator. □

We finally get our main theorem that relates the absence of orbits in the leakage diagram to the secret independence.

Theorem 1 (Secret Independence from Absence of Orbits). *Let $n \in \mathbb{N}$, let C be an arithmetic circuit, and let C_n be its compiled version (Definition 5). Let Φ be a mapping associated with the multiplication gadgets used in the compiler such that for every multiplication gadget G in C_n and every leakage set $\mathcal{W}_G$ of wires of G, the gadget G is $(\Phi(\mathcal{W}_G), \mathcal{W}_G)$-generalized free-SNI. Let $\mathcal{W}$ be a set of wires of C_n.*

If the $(\mathcal{W}, \Phi, C_n)$-leakage diagram contains no orbit, then C_n is secret independent for $\mathcal{W}$ with respect to the uniform additive encoding.

Proof. From Proposition 3, if there is no orbit, then the covering with respect to each input sharing is incomplete. Further, all the gadgets in the composition are covering simulatable: by Proposition 5 for the refresh, and from ORS for the other gadgets, through Proposition 6 (ORS itself comes from Lemma 2 for multiplication gadgets and from Proposition 2 for the other gadgets). Thanks to the composability of covering simulatability (Proposition 4), the whole C_n is simulatable by its covering. Finally, since we have proven that $\mathcal{W}$ is simulatable by incomplete sets of input share intervals, the secret independence directly follows from Proposition 1. □

4.4 Random Probing Security from Absence of Orbits

Having established that the absence of orbits in the leakage diagram associated with a leakage set $\mathcal{W}$ implies the secret independence of the compiled circuit, we now turn to the random probing model. We will bound the probability that the leakage diagram contains an orbit, which corresponds to the probability of simulation failure and therefore yields an upper bound on the random probing advantage.

First, we reformulate leakage diagrams directly in the random probing setting and show that this probabilistic definition coincides with Definition 7 when the leakage set $\mathcal{W}$ is instantiated as LeakingWires.

Definition 18 (Random Probing Leakage Diagram). *Let $n \in \mathbb{N}$ and $p \in [0,1]$. Let C be an arithmetic circuit, let C_n be its compiled circuit as in Definition 5, and let Γ_{C_n} be its associated dependency graph (Definition 6). Let Φ be a mapping assigning to any subset of wires of a multiplication gadget G a subset of indices in $[0, n-1]$ such that for all sets of wires $\mathcal{W}$, G is $(\Phi(\mathcal{W}), \mathcal{W})$-generalized free SNI. The* (C_n, Φ, p)-random probing leakage diagram *is the random variable corresponding to the drawing of random subgraph $\Gamma^p_{C_n} \subseteq \Gamma_{C_n}$ defined as follows.*

- *Refresh gadgets. Each r-edge of Γ_{C_n} is added to $\Gamma^p_{C_n}$ with probability $1-(1-p)^3$ and each b-edge is added with probability p.*
- *Affine gadgets. Each ν_i^G edge in Γ_{C_n} for G an affine gadget is added to $\Gamma^p_{C_n}$ with probability $1-(1-p)^t$, where t is the number of wires associated with one share index in G.*
- *Multiplication gadgets. Each set $(\nu_i^G)_{i \in I}$ of edges in Γ_{C_n} for a multiplication gadget G belongs to $\Gamma^p_{C_n}$ with the probability*

$$\Pr[\Phi(\mathcal{W}) = I \mid \mathcal{W} \leftarrow \mathsf{LeakingWires}(G, p)] \,.$$

In practice, a natural choice for the mapping Φ is to associate to each leakage set $\mathcal{W}$ the smallest set I such that the considered multiplication gadget is $(I, \mathcal{W})$-generalized free SNI. Ultimately, we get the random probing security from the probability of orbits in the random probing leakage diagram.

Corollary 1 (RP Security). *Let $n \in \mathbb{N}$ and $p \in [0,1]$. Let C be an arithmetic circuit, let C_n be its compiled circuit, and let Φ be a mapping such that every multiplication gadget G of C_n is $(\Phi(\mathcal{W}), \mathcal{W})$-generalized free-SNI for every leakage set $\mathcal{W}$ of wires of G. Let $\Gamma^p_{C_n}$ be the (C_n, Φ, p)-random probing leakage diagram of Definition 18, and let*

$$\varepsilon = \Pr\big[\Gamma^p_{C_n} \text{ contains an orbit}\big] .$$

Then C_n is (p, ε)-random probing secure with respect to the uniform additive encoding.

Proof (sketch). The distribution of the leakage diagram $\Gamma^{\mathcal{W}}_{C_n, \Phi}$ with $\mathcal{W}$ sampled from $\mathsf{LeakingWires}$ coincides with the random probing diagram, and the leakage is secret independent where there is no orbit by Theorem 1. □

5 Dominating Probabilities for Dependency Edges

In the previous section, we showed that the absence of orbits in the random probing leakage diagram implies the secret independence of the compiled circuit. More precisely, each gadget of the compiler contributes dependency edges to the leakage diagram according to the random probing leakage affecting its internal wires. In this section, we focus on the horizontal dependency edges in the diagram. Since the input and output share leakage of the refresh gadgets is accounted for by the surrounding gadgets, the horizontal dependency edges are induced by the non-refresh gadgets of the compiler, namely affine and multiplication gadgets. Our goal is to derive upper bounds on the probability that several dependency edges for a gadget simultaneously appear in the random probing leakage diagram. These bounds will later be used in Section 6 to estimate the probability that the leakage diagram contains an orbit, along with the distribution of the edges for the refresh gadgets given in Definition 18.

5.1 Generic Framework for Dependency Edge Distributions

Let G be a non-refresh gadget of the compiler (*i.e.*, multiplication or affine) with n shares, and let $\mathcal{W}_G$ denote its set of wires. Let us recall that $(\nu_i^G)_{i \in [n]}$ denote the dependency edges associated with the gadget G in the leakage diagram, we define the random variable $\mathcal{I}_G \triangleq \{i \in [n] : \nu_i^G \in \Gamma^p_{C_n}\}$, where $\Gamma^p_{C_n}$ is the random probing leakage diagram.

The orbit analysis of Section 6 bounds the probability of having an orbit in the global random probing leakage diagram based on a simple characterization of the individual gadgets: a bound on the number of dependency edges appearing simultaneously in each gadget.

Definition 19 (Domination). *Let* $\boldsymbol{p} = (p^{(0)}, \ldots, p^{(n)}) \in \mathbb{R}^{n+1}$*, and let* $\mathcal{I}$ *be a random variable such that* $\mathcal{I} \subset [n]$*. We say that* $\boldsymbol{p}$ *dominates* $\mathcal{I}$ *if, for every* $w \in \{0, \ldots, n\}$ *and every* $S \subset [n]$ *with* $|S| = w$*,* $\Pr[S \subset \mathcal{I}] \leq p^{(w)}$*.*

5.2 Affine Gadgets

We now instantiate the generic framework of the previous subsection for affine gadgets. Unlike multiplication gadgets, affine gadgets do not create interactions between different share indices: each dependency edge only depends on the leakage affecting the corresponding share.

For an affine gadget G_{aff}, let t_{aff} be the number of wires associated with one share index, the set of dependency edges in the random probing diagram $\mathcal{I}_{G_{\text{aff}}}$ is such that for any $i \in [n]$, $\Pr[i \in \mathcal{I}_{G_{\text{aff}}}] = 1 - (1-p)^{t_{\text{aff}}}$, by Definition 18. Since affine gadgets process each share independently, the events $(i \in \mathcal{I}_{G_{\text{aff}}})_{i \in [n]}$ are jointly independent, therefore, we get the tight dominating probabilities

$$p^{(w)}_{G^{t_{\text{aff}}}_{\text{aff}}} \triangleq \left(1 - (1-p)^{t_{\text{aff}}}\right)^w$$

In the following, we will only consider three types of affine gadgets: addition gadgets, for which $t_{\text{aff}} = 3$; copy gadgets, for which $t_{\text{aff}} = 3$ under the copy convention of [14]; and affine 1-to-1 gadgets, for which $t_{\text{aff}} = 2$.

5.3 Multiplication Gadgets

Let us now compute the dominating probabilities for the multiplication gadgets.

Dependency mapping. For the multiplication gadget G, Definition 18 requires a mapping Φ_G such that G is $(\Phi_G(\mathcal{W}), \mathcal{W})$-generalized free-SNI for any $\mathcal{W} \subset \mathcal{W}_G$. The dependency edges $\mathcal{I}_G$ that appear in the random probing leakage diagram are then $\mathcal{I}_G = \Phi_G(\mathsf{LeakingWires}(G, p))$.

Multiplication gadget of [14]. We now instantiate this generic domination and dependency mapping framework with the multiplication gadget of [14], which we denote by G_{mult} and recalled in Algorithm 2. Using the dependency mapping introduced in the original analysis of [14], we refine their random probing analysis through the dominating probabilities $(p^{(w)}_{G_{\text{mult}}})_{w \in [n]}$.

Our first improvement already appears at the level of the single-edge dominating probability $p^{(1)}_{G_{\text{mult}}}$. The original analysis of [14] upper bounds the probability that a multiplication dependency edge appears by

$$1 - (1-p)^{8n} + 1 - (1 - \sqrt{3p})^{n-1}.$$

By refining the counting of the wires associated with each dependency set, we improve this bound to

$$1 - (1-p)^{6n} + 1 - (1 - \sqrt{3p})^{n-1}.$$

More generally, our framework also yields bounds on the probabilities $p^{(w)}_{G_{\text{mult}}}$ for arbitrary values of w. Unlike [14], these bounds preserve the correlations between multiplication dependency edges instead of reducing the leakage distribution to an independent edge model. This will later lead to tighter estimates for the probability that the global leakage diagram contains an orbit.

In the original analysis of [14], the dependency edge $\nu_i^{G_{\text{mult}}}$ is added to the leakage diagram whenever at least one variable from the set

$$T_i^{\text{mult}} = \{a_i, b_i, a_i \cdot b_j, a_j \cdot b_i, w_{i,j}, w_{j,i}, z_{\ell,i,j} | j \in [n], \ell \in [\log_2(n+1) + 1]\}$$

leaks, for some $i \in [n]$. Accordingly, for every leakage set $\mathcal{W}$, we define

$$\Phi_{G_{\text{mult}}}(\mathcal{W}) = \{i \in [n] \mid \mathcal{W} \cap T_i^{\text{mult}} \neq \emptyset\}.$$

The following proposition shows that this mapping is sound with respect to the generalized free SNI property. The proof is given in Appendix C.1.

Algorithm 2 Multiplication gadget G_{mult} from [14]

Input: two n-sharings $[\![a]\!] = (a_0, \ldots, a_{n-1})$ and $[\![b]\!] = (b_0, \ldots, b_{n-1})$.
Output: an n-sharing $[\![c]\!] = (c_0, \ldots, c_{n-1})$ such that $\sum_{i=0}^{n-1} c_i = \left(\sum_{i=0}^{n-1} a_i\right)\left(\sum_{i=0}^{n-1} b_i\right)$.

for $i = 1$ to $n-1$ **do**
  for $j = 0$ to $i-1$ **do**
    $z_{0,i,j} \xleftarrow{\$} \mathbb{K}$
for $i = 0$ to $n-1$ **do**
  for $j = i+1$ to $n-1$ **do**
    $w_{i,j} \leftarrow a_i b_j - z_{0,j,i}$
    $z_{0,i,j} \leftarrow w_{i,j} + a_j b_i$
for $i = 0$ to $n-1$ **do**
  $z_{0,i,i} \leftarrow a_i b_i$
$L \leftarrow \log_2(n+1)$
for $i = 0$ to $n-1$ **do**
  for $\ell = 1$ to L **do**
    for $j = 0$ to $2^{L-\ell} - 1$ **do**
      $z_{\ell,i,j} \leftarrow z_{\ell-1,i,2j} + z_{\ell-1,i,2j+1}$
for $i = 0$ to $n-1$ **do**
  $c_i \leftarrow z_{L,i,0}$
return $[\![c]\!]$

Proposition 7. *Let $\mathcal{W}$ be a set of wires of G_{mult}. Then G_{mult} is*

$$(\Phi_{G_{\text{mult}}}(\mathcal{W}), \mathcal{W})\text{-generalized free SNI.}$$

We now analyze the dominating probabilities associated with G_{mult}. For every $i \in [n]$, the edge $\nu_i^{G_{\text{mult}}}$ appears in the leakage diagram if and only if at least one wire belonging to T_i^{mult} leaks. To ensure a fair comparison with [14], we adopt the same convention for copy gates, that is for all $x \in \mathbb{K}$ and $m \geq 2$, when x is copied into m copies, it is carried by $m+1$ wires. Under this convention, we first derive an upper bound on the probability of having a given edge in the leakage diagram.

Lemma 3 (Single-Edge Probability). *Let $n \in N$ and $p \in [0,1]$. For every $i \in [n]$, we have*

$$p_{G_{\text{mult}}}^{(1)} \leq 1 - (1-p)^{6n} + 1 - (1 - \sqrt{3p})^{n-1}.$$

The proof is given in Appendix C.2. It follows the same approach as [14], namely by counting the wires involved in each event and enforcing their independence, but yields tighter bounds through a more refined counting analysis.

Using this first bound, we can now derive dominating probabilities for G_{mult}. A proof sketch is given below, while the full proof is deferred to Appendix C.3.

Lemma 4 (Multi-Edge Dominating Probability). *Let $n \in N$ and $p \in [0,1]$. For every $w \in \{0, \ldots, n\}$, let*

$$p_{G_{\text{mult}}}^{(w)} \triangleq 1 + \sum_{\ell=1}^{w} \binom{w}{\ell} (-1)^{\ell} (1-p)^{\frac{3\ell}{2}(6n-\ell-1)}$$

then $\boldsymbol{p}_{G_{\text{mult}}}$ dominates $\Phi_{G_{\text{mult}}}(\mathsf{LeakingWires}(G_{\text{mult}}, p))$.

Proof (sketch). The result follows from an inclusion-exclusion argument on the events $\{Z_i = 1\}_{i \in [n]}$, where Z_i denotes the event that the edge $\nu_i^{G_{\text{mult}}}$ is added to the leakage diagram. Using Lemma 3, an edge can be added either exclusively with probability $1 - (1-p)^{6n}$ or jointly with another edge with probability $1 - (1-p)^3$. Since each multiplication variable belongs to at most two sets T_i, T_j, and the underlying wire leakages are independent, the probability decomposes according to the number of internal and external pairwise interactions, yielding the claimed expression after grouping terms by cardinality. □

5.4 Gadget-agnostic domination

For the analysis in the next section, we want a single $\boldsymbol{p}_{\text{all}}$ that dominates the dependency edge distributions of all our non-refresh gadgets. Since affine gadgets satisfy $t_{\mathsf{aff}} \leq 3$, we can use $p_{\text{all}}^{(w)} \triangleq \max\left(p_{G_{\text{mult}}}^{(w)}, p_{G_{\text{aff}}^3}^{(w)}\right)$, for every $w \in [n+1]$.

Further, we also want that for any w_1 and w_2, $p^{(w_1+w_2)}$ is an upper-bound on the probability that the leakage diagram contains w_1 dependency edges for one gadget and w_2 dependency edges for another gadget. Since the $\mathcal{I}_G$ of independent gadgets are independent of each other, we can take

$$p_{\max}^{(w)} \triangleq \max_{\substack{w_1,w_2\geq 0 \\ w_1+w_2=w}} p_{\text{all}}^{(w_1)} p_{\text{all}}^{(w_2)}.$$

For readability purposes, in the remainder of the paper we simply denote $p_{\max} \triangleq p_{\max}^{(1)}$.

6 Tighter Security Bound for the Compiler

We aim to obtain a tighter bound than the one provided in [14, Theorem 4] on the random probing security (RPS) advantage of the compiler of Definition 5. By Corollary 1, this advantage corresponds to the probability that the leakage diagram contains at least one orbit. As shown in [14], any orbit has size at least n. The following theorem then follows from a union bound over all possible orbit sizes.

Theorem 2 (Orbit decomposition of the RPS advantage). *Let $n \in \mathbb{N}$, let C be an arithmetic circuit, and let C_n be its compiled circuit. Let Φ be a mapping associated with the multiplication gadgets of $\mathcal{C}$ such that every multiplication gadget G of $\mathcal{C}$ is $(\Phi(\mathcal{W}_G), \mathcal{W}_G)$-generalized free-SNI for every leakage set $\mathcal{W}_G$ of wires of G. Let $\Gamma^p_{C_n,\Phi}$ be the corresponding random probing leakage diagram. With the above notations, we have*

$$\begin{aligned}
\varepsilon_{\text{RPS}}(\mathcal{C}, p) &= \Pr\left[\Gamma^p_{C_n,\Phi} \text{ contains an orbit}\right] \\
&\leq \Pr\left[\Gamma^p_{C_n,\Phi} \text{ contains an orbit of size } n\right] \\
&\quad + \Pr\left[\Gamma^p_{C_n,\Phi} \text{ contains an orbit of size } n+1\right] \\
&\quad + \Pr\left[\Gamma^p_{C_n,\Phi} \text{ contains an orbit of size } n+2\right] \\
&\quad + \Pr\left[\Gamma^p_{C_n,\Phi} \text{ contains an orbit of size at least } n+3\right].
\end{aligned}$$

6.1 Generic Properties on Orbits

First, we show that any orbit necessarily contains an even number of non-horizontal edges (*i.e.*, edges of type r or b). Next, we derive a lower bound on the number of such non-horizontal edges depending on the orbit size. The proofs of the following propositions are detailed in Appendix D.1 and Appendix D.2.

Proposition 8 (Even Number of Non-Horizontal Edges). *Let O be an orbit of size k in $\Gamma^p_{C_n,\Phi}$, with $k \geq n$. Then O contains an even number of non-horizontal edges (*i.e.*, refresh edges of type b or r).*

Proposition 9 (Minimum Number of Non-Horizontal Edges). *Let O be an orbit of size k in $\Gamma^p_{C_n,\Phi}$. Let $i \in [1, \lfloor k/2 \rfloor]$ and assume $k \geq n + 3i - 2$. Then O contains at least $2i$ non-horizontal edges (i.e., edges of type b or r).*

6.2 Computation of Intermediate Terms

We now explicitly expand the four terms of Theorem 2 corresponding to orbits of sizes n, $n+1$, $n+2$, and at least $n+3$.

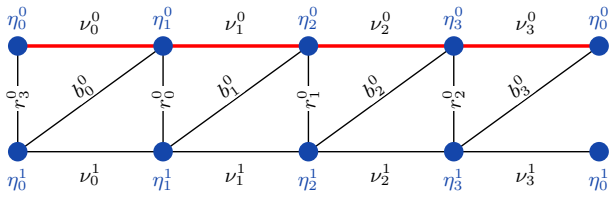


Fig. 5: Example of orbit (in red) of size n

Orbits of Size n. We first consider orbits of minimal size n. By construction, any non-horizontal edge (*i.e.*, a refresh edge of type b or r) forces the orbit to leave the circular structure of a gadget, thereby increasing its size. Consequently, orbits of size exactly n can only occur when the orbit is entirely contained within a single gadget and uses only horizontal edges. Equivalently, such an orbit exists if and only if all edges of a circular layer of a non-refresh gadget are present in the leakage diagram, as illustrated in red Figure 5 for $n = 4$. This yields the following lemma.

Lemma 5 (Orbits of Size n). *The probability that $\Gamma^p_{C_n,\Phi}$ contains at least one orbit of size n is*

$$\Pr\left[\Gamma^p_{C_n,\Phi} \text{ contains an orbit of size } n\right] = 1 - \prod_{\alpha\in\{\text{lin,add,mult,copy}\}} \left(1-p_\alpha^n\right)^{|\mathcal{G}_\alpha^{C_n}|},$$

where p_α denotes the probability for an edge of type α to be in $\Gamma^p_{C_n,\Phi}$.

Proof. By the discussion preceding the lemma, an orbit of size exactly n can only be entirely contained in a single non-refresh gadget. Therefore, the event $[\Gamma^p_{C_n,\Phi}$ contains an orbit of size $n]$ is the union, over all non-refresh gadgets G in C_n, of the event that all n horizontal edges of G belong to the leakage diagram $\Gamma^p_{C_n,\Phi}$. For each such gadget G, let E_G denote the event that G gives rise to an orbit of size n, *i.e.*, that all its n horizontal edges belong to $\Gamma^p_{C_n,\Phi}$. Since distinct non-refresh gadgets involve disjoint sets of wires, and wires leak independently in the random probing model, the events $(E_G)_G$ are independent. Hence,

$$\Pr\left[\Gamma^p_{C_n,\Phi} \text{ contains an orbit of size } n\right] = \Pr\left[\bigcup_G E_G\right] = 1 - \Pr\left[\bigcap_G \overline{E_G}\right] = 1 - \prod_G \Pr\left[\overline{E_G}\right].$$

Now, if G is of type $\alpha \in \{\text{lin}, \text{add}, \text{mult}, \text{copy}\}$, the event E_G occurs exactly when its n horizontal edges all belong to $\Gamma^p_{C_n,\Phi}$, which happens with probability p_α^n. Therefore, $\Pr\left[\overline{E_G}\right] = 1 - p_\alpha^n$. □

Orbits of Size $n+1$. For orbits of size $n+1$, we distinguish two families. The first family consists of orbits that use edges of the following refresh gadget and may contain a downward plateau, *i.e.*, one horizontal edge in the next gadget in the execution order (first term of the next lemma; see the red path on Figure 6 for the case $n = 4$). The second family consists of orbits that move upward towards the preceding gadget, without any plateau (second term of the next lemma; see the green path of Figure 6 for the case $n = 4$). The proof is given in Appendix D.3.

Lemma 6 (Orbits of Size $n+1$). *The probability that $\Gamma^p_{C_n,\Phi}$ contains at least one orbit of size $n+1$ is upper bounded by*

$$\begin{aligned}
&\Pr\left[\Gamma^p_{C_n,\Phi} \text{ contains an orbit of size } n+1\right] \\
&\leq 3\cdot p^2 \cdot \sum_{b=0}^{n-1}\sum_{r=b}^{n+b-2} p_{\max}^{(r-b)} \cdot \left(\sum_{\alpha\in\{\text{add,mult,lin}\}} p_\alpha^{n-1-(r-b)}\cdot|\mathcal{G}_\alpha| \;+\; 2\cdot p_{\text{copy}}^{n-1-(r-b)}\cdot|\mathcal{G}_{\text{copy}}|\right) \\
&\quad + 3\cdot p^2\cdot n\cdot\left(2\cdot p_{\text{mult}}^{n-1}\cdot|\mathcal{G}_{\text{mult}}| + 2\cdot p_{\text{add}}^{n-1}\cdot|\mathcal{G}_{\text{add}}| + p_{\text{copy}}^{n-1}\cdot|\mathcal{G}_{\text{copy}}| + p_{\text{lin}}^{n-1}\cdot|\mathcal{G}_{\text{lin}}|\right),
\end{aligned}$$

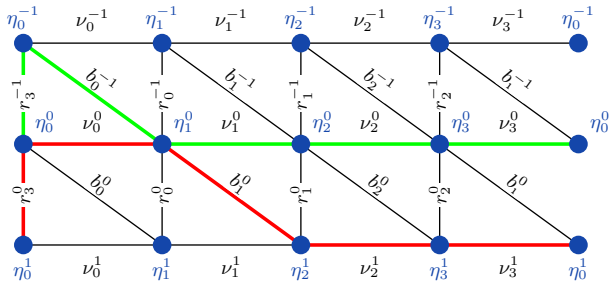


Fig. 6: Examples of two orbit paths: the lower red path illustrates the first term with a possible plateau, whereas the upper green path illustrates the second term for which no plateau is allowed.

where $\mathcal{G}_{\text{lin}}$, $\mathcal{G}_{\text{add}}$, $\mathcal{G}_{\text{mult}}$, *and* $\mathcal{G}_{\text{copy}}$ *denote the sets of gadgets of the corresponding types (linear* 1*-to-*1*, addition, multiplication, and copy) in* C_n.

Orbits of Size $n+2$. We now turn to orbits of size $n+2$. In this case, several configurations may arise, depending on how the orbit deviates from the circular structure of a single gadget. More precisely, we distinguish cases according to whether the orbit reaches one or two preceding gadgets, one or two subsequent gadgets, or a neighboring gadget together with an alternative input/output (in the case of multi-input/output gadgets), as well as according to the types of non-horizontal edges used to move between gadgets.

Lemma 7. *The probability that* $\Gamma^p_{C_n,\Phi}$ *contains at least one orbit of size* $n+2$ *is upper bounded as follows*

$$
\begin{aligned}
&\Pr\Big[\Gamma^p_{C_n,\Phi} \text{ contains an orbit of size } n+2\Big] \\
&\le 10p^2 \sum_{d=1}^{n-1} (n-d)\Big(p_{\max}^{(d)}\Phi_2(n-d) + p_{\max}^{(n-d)}\Phi_2(d)\Big) \\
&\quad + (3p^2)^2\Bigg[n\sum_{\Delta=1}^{n-2}(n-\Delta)(n-\Delta-1)p_{\max}^{(n-2-\Delta)}\Phi_2(\Delta) + 2n\Psi(n-2) \\
&\qquad + \sum_{(r^0,b^0,r^1,b^1)\in\mathcal{A}_1} p_{\max}^{(n-b^0+r^0-b^1+r^1)}\Phi_4(b^0-r^0+b^1-r^1-2) \\
&\qquad + \sum_{(b^0,r^0,b^1,r^1)\in\mathcal{A}_2} p_{\max}^{(r^0-b^0+r^1-b^1)}\Phi_4(n-2-(r^0-b^0+r^1-b^1)) \\
&\qquad + \binom{n}{2}\Phi_2(n-2) + n^2 p_{\text{copy}}^{n-2}|\mathcal{G}_{\text{copy}}| + n(n-1)\Theta(n-2) + n\sum_{d=3}^{n-1} p_{\max}^{(d-2)}\Psi(n-d)\Bigg],
\end{aligned}
$$

$$
\begin{aligned}
\textit{where } \Phi_\gamma(t) &= \sum_{\alpha\in\{\text{add},\text{mult},\text{lin}\}} p_\alpha^t|\mathcal{G}_\alpha| + \gamma\, p_{\text{copy}}^t|\mathcal{G}_{\text{copy}}|, \\
\Psi(t) &= 2p_{\text{mult}}^t|\mathcal{G}_{\text{mult}}| + 2p_{\text{add}}^t|\mathcal{G}_{\text{add}}| + p_{\text{copy}}^t|\mathcal{G}_{\text{copy}}| + p_{\text{lin}}^t|\mathcal{G}_{\text{lin}}|, \\
\Theta(t) &= 2\sum_{\alpha\in\{\text{add},\text{mult},\text{copy}\}} p_\alpha^t|\mathcal{G}_\alpha| + p_{\text{lin}}^t|\mathcal{G}_{\text{lin}}|,
\end{aligned}
$$

$$
\textit{and } \mathcal{A}_1 = \Big\{(r^0,b^0,r^1,b^1) : 0\le r^0 < b^0 \le r^1 < b^1 < n,\ (b^0,b^1)\neq(r^0+1,r^1+1)\Big\},
$$
$$
\mathcal{A}_2 = \Big\{(b^0,r^0,b^1,r^1) : 0\le b^0 \le r^0 < b^1 \le r^1 < n,\ (b^0,b^1,r^1)\neq(0,r^0+1,n-1)\Big\}.
$$

The full proof is given in Appendix D.4. We only sketch here the possible configurations leading to orbits of size $n+2$. Such orbits may arise from two b-edges (or two r-edges) within the same downward gadget, from two successive downward or upward traversals, from traversals involving one upward and one downward gadget, or from configurations involving copy gadgets and their secondary inputs or outputs. Depending on the case, these configurations may create a plateau or span one or two neighboring gadgets. Together, they exhaust all possible ways an orbit of size $n+2$ can deviate from the circular structure induced by the leakage diagram with circular refresh gadgets.

Orbits of Size at least $n+3$. We have derived upper bounds for the first three terms of Theorem 2, corresponding to orbits of sizes n, $n+1$, and $n+2$. It remains to control the contribution of larger orbits, namely the probability that the leakage diagram contains at least one orbit of size at least $n+3$.

Lemma 8. *Let β be a constant that can be adjusted for more precision. The probability that $\Gamma^p_{C_n,\Phi}$ contains at least one orbit of size at least $n+3$ is upper bounded by*

$$\begin{aligned}
&\Pr\Big[\Gamma^p_{C_n,\Phi} \text{ contains an orbit of size at least } n+3\Big] \\
&\le n(3p^2)\sum_{\Delta=1}^{n-2} p_{\max}^{(n-\Delta)}\left(\sum_{\alpha\in\{\text{add,mult,lin}\}} p_\alpha^{\Delta+1}|\mathcal{G}_\alpha| + 2p_{\text{copy}}^{\Delta+1}|\mathcal{G}_{\text{copy}}|\right) \\
&\quad + \sum_{k=n+3}^{\beta}\sum_{i=\max\left(\left\lceil\frac{k-n}{3}\right\rceil,2\right)}^{\lfloor k/2\rfloor}\Bigg[|\mathcal{G}_{\text{mult}}|\binom{k}{2i}(12p)^{2i}(2p_{\text{mult}})^{k-2i} \\
&\qquad + \sum_{L=\max(2i,k-n)}^{k}(k-L+1)\binom{L-2}{2i-2}(2p_{\max})^{L-2i} \\
&\qquad\quad \cdot\Big((12p)^{2i}\big(|\mathcal{G}_{\text{add}}|(2p_{\text{add}})^{k-L} + |\mathcal{G}_{\text{copy}}|(2p_{\text{copy}})^{k-L}\big) \\
&\qquad\qquad + (8p)^2(12p)^{2i-2}|\mathcal{G}_{\text{lin}}|(2p_{\text{lin}})^{k-L}\Big)\Bigg] + |\mathcal{G}|\,(12p+2p_{\max})^{\beta+1}.
\end{aligned}$$

Proof (sketch). We only give the main counting argument here; the complete proof is deferred to Appendix D.5. A naive upper bound on the probability that there exists an orbit of size at least $n+3$ can be obtained by counting all possible edge combinations, yielding $|\mathcal{G}|\cdot(12\cdot p+2\cdot p_{\text{mult}})^{n+3}$ but this is overly pessimistic since many edge configurations cannot form valid orbits. We only use it for orbits of size $\beta+1$ and more.

Let $E_{k,i}$ be the event that $\Gamma^p_{C_n,\Phi}$ contains an orbit of size k with exactly $2i$ non-horizontal edges. Using Proposition 8 and Proposition 9, we obtain

$$\begin{aligned}
&\Pr\Big[\Gamma^p_{C_n,\Phi} \text{ contains an orbit of size at least } n+3\Big] \\
&\le \Pr[E_{n+3,1}] + \sum_{k=n+3}^{\beta}\sum_{i=\max(\lceil(k-n)/3\rceil,2)}^{\lfloor k/2\rfloor}\Pr[E_{k,i}] + |\mathcal{G}|(12p+2p_{\max})^{\beta+1}.
\end{aligned}$$

The first term corresponds to orbits of size $n+3$ with exactly two non-horizontal edges. Counting these orbits gadget by gadget (as in the cases $n+1$ and $n+2$) yields

$$\Pr[E_{n+3,1}] \le n(3p^2)\sum_{\Delta=1}^{n-2} p_{\max}^{(n-\Delta)}\left(\sum_{\alpha\in\{\text{add,mult,lin}\}} p_\alpha^{\Delta+1}|\mathcal{G}_\alpha| + 2p_{\text{copy}}^{\Delta+1}|\mathcal{G}_{\text{copy}}|\right).$$

For the remaining terms, we count all placements of the $2i$ non-horizontal edges. A coarse bound is $|\mathcal{G}|\binom{k}{2i}(12p)^{2i}(2p_{\max})^{k-2i}$. We then refine this estimate gadget by gadget. For addition and copy gadgets, horizontal edges inside the gadget contribute probabilities p_{add} or p_{copy} instead of $p_{\max}$. For linear gadgets, the exit and return edges contribute $(8p)^2$ instead of $(12p)^2$. Summing over the possible positions of the first exit and last return gives the refined expression stated in the lemma. □

Remark 2. We partially validated the above bounds (for orbits of size $n+1$ and $n+2$) using the verification tool IronMask [9] in the case of consecutive linear (1-to-1) gadgets. To this end, we implemented the corresponding configurations and printed the number of incompressible tuples. The results are in exact agreement with our derived expressions.

7 Application to AES and Comparison

In this section, we apply our compiler to an AES circuit and leverage our results from Lemma 3 and Lemma 4 (on multiplication), together with Theorem 2 and the bounds derived for each orbit size, to obtain a lower bound on the security achieved in the random probing model. Our code is publicly accessible at `https://github.com/CryptoExperts/AC26-leakage-diagram-rp-compiler`.
We compare our bound with the following state-of-the-art approaches:

1. The original work of Berti, Faust, and Orlt [14] that derives random probing security bounds using the same compiler framework. As in this paper, the security is expressed as a function of the number of shares n and the leakage rate p.
2. The approach of Jahandideh, Mennink and Batina [23] which does not use a simulation-based random probing security; instead, it provides a bound on the mutual information between the leakage and a single byte of the secret key, resulting in random probing security bounds. The security bound is not formally proven, it is a function of n and p extrapolated from tool estimations.
3. The work of Belaïd, Normand, and Rivain [10] that introduces a new multiplication gadget with cardinal random probing composability, enabling tighter security guarantees. Notably, it allows reaching very high security levels such as 2^{-64}, 2^{-80}, and 2^{-128}.
4. The work of Belaïd and Cassiers [7] that introduces a tool for direct random probing security estimation without going through a composition theorem. We compare the result on their tool PERSEUS on their implementation of the AES, with their instantiation of "ISW" multiplication gadget, since that gadget is the most similar to the one we use in this work and has the same cost.

The comparison is illustrated in Figure 7 for leakage rates $p = 2^{-12}$ and $p = 2^{-16}$. We observe that our approach provides a significant refinement over the security bounds of [14] in the considered settings. In particular, for $p = 2^{-16}$, we achieve the security level 2^{-40} for $n = 8$, whereas this level is not reached in [14] for this leakage rate, even with many shares. For $p = 2^{-12}$, we can reach high security levels in practical numbers of shares, whereas the bound of [14] provides no security.

Comparing to other approaches, we first observe that our bounds are also generally a bit tighter than the extrapolation of [23] in most cases, while being provably correct bounds.

With respect to the compiler of [10], our work often requires a few more shares in order to achieve the same security level (e.g., for $p = 2^{-16}$ and a target security level of 2^{-64}, $n = 10$ vs $n = 13$ shares). This gap becomes more pronounced as the number of shares increases, and likely originates from their more efficient multiplication gadget, which provides a better trade-off for random probing security. However, our construction generally achieves a better performance vs security trade-off. For example, at 2^{-64} security with $p = 2^{-16}$, the compiler of [10] requires $255k$ random gates, whereas we only need $189k$, despite the higher share count.

Finally, for a given number of shares, the bounds of [7] are better than the ones from this work, and their computational complexity is very close to ours. However, their approach suffers from two main limitations. First, their verification based on a Monte Carlo algorithm becomes prohibitively costly to run for higher security levels. Second, since it is not based on composition, it will be expensive to run on very large circuits, such as the ones of public-key cryptographic algorithms, unlike our bounds.

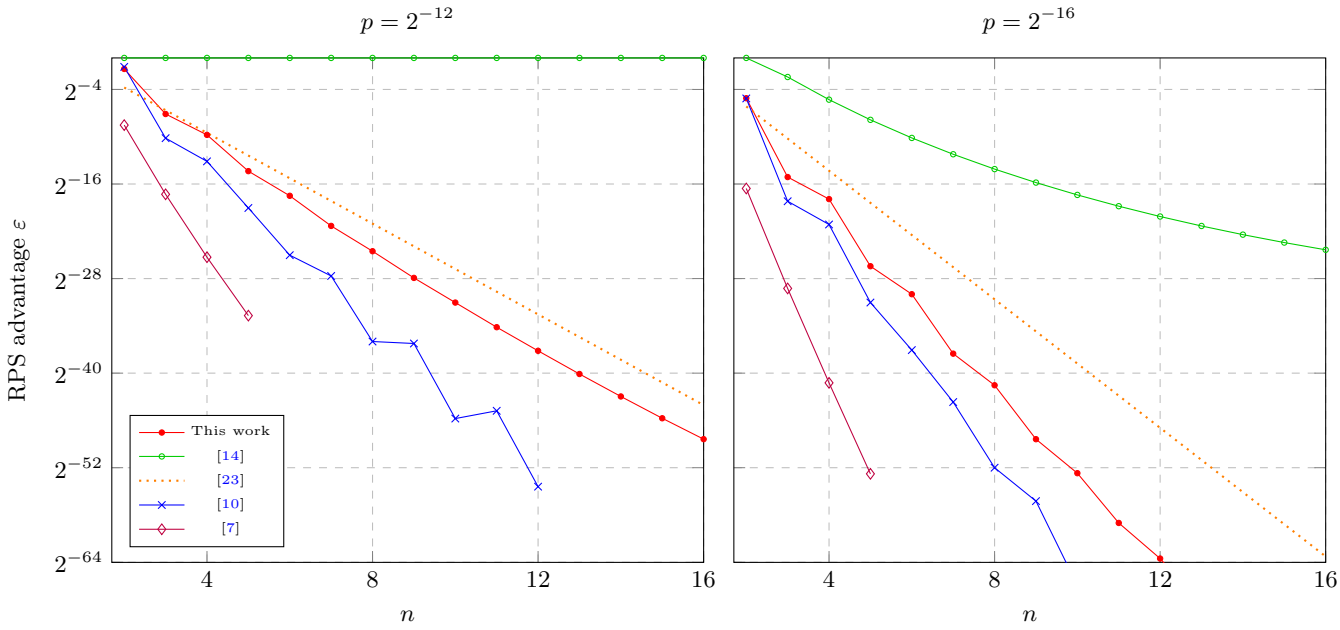


Fig. 7: Random Probing Security of masked AES with state of the art compilers

8 Conclusion

Starting from the work of Berti, Faust and Orlt [14], we revisited the security analysis based on a more precise formalization of the leakage diagrams. Unlike the original work, we follow an approach based on a simulatability definition, which allows us to keep the link between random probing security and orbits in the diagram, while introducing a simple assumption for the properties of the (multiplication) gadgets: generalized free-SNI. For this purpose, and perhaps of independent interest, we generalize the classic notion of simulatability, giving partial sums of input shares to the simulator. We then provide a finer-grained analysis of the multiplication gadget, and a tighter count of the orbits in the diagram.

Overall, our bounds represent a significant improvement over [14], and show that leakage diagrams are competitive with other state-of-the-art methods for proving random probing security.

As future work, it would be particularly interesting to integrate the multiplication gadget of [10] into the leakage-diagram framework by proving that it satisfies the generalized free SNI property introduced in this work. Given the improved security/performance trade-offs already observed in [10], combining this multiplication gadget with our refined leakage-diagram analysis could further improve the resulting random probing security bounds.

Acknowledgments. We warmly thank Maximilian Orlt for his helpful answers to our questions. This work was co-funded by the European Union (ERC, AMAskZONE, 101077506). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

References

1. Miklós Ajtai. Secure computation with information leaking to an adversary. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 715–724. ACM Press, June 2011.

2. Prabhanjan Ananth, Yuval Ishai, and Amit Sahai. Private circuits: A modular approach. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 427–455. Springer, Cham, August 2018.
3. Marcin Andrychowicz, Stefan Dziembowski, and Sebastian Faust. Circuit compilers with $O(1/\log(n))$ leakage rate. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 586–615. Springer, Berlin, Heidelberg, May 2016.
4. Gilles Barthe, Sonia Belaïd, François Dupressoir, Pierre-Alain Fouque, Benjamin Grégoire, Pierre-Yves Strub, and Rébecca Zucchini. Strong non-interference and type-directed higher-order masking. In Edgar R. Weippl, Stefan Katzenbeisser, Christopher Kruegel, Andrew C. Myers, and Shai Halevi, editors, *ACM CCS 2016*, pages 116–129. ACM Press, October 2016.
5. Gilles Barthe, François Dupressoir, Sebastian Faust, Benjamin Grégoire, François-Xavier Standaert, and Pierre-Yves Strub. Parallel implementations of masking schemes and the bounded moment leakage model. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part I*, volume 10210 of *LNCS*, pages 535–566. Springer, Cham, April / May 2017.
6. Alberto Battistello, Jean-Sébastien Coron, Emmanuel Prouff, and Rina Zeitoun. Horizontal side-channel attacks and countermeasures on the ISW masking scheme. In Benedikt Gierlichs and Axel Y. Poschmann, editors, *CHES 2016*, volume 9813 of *LNCS*, pages 23–39. Springer, Berlin, Heidelberg, August 2016.
7. Sonia Belaïd and Gaëtan Cassiers. PERSEUS - probabilistic evaluation of random probing security using efficient sampling. In *EUROCRYPT (7)*, Lecture Notes in Computer Science, pages 506–536. Springer, 2026.
8. Sonia Belaïd, Jean-Sébastien Coron, Emmanuel Prouff, Matthieu Rivain, and Abdul Rahman Taleb. Random probing security: Verification, composition, expansion and new constructions. In Daniele Micciancio and Thomas Ristenpart, editors, *CRYPTO 2020, Part I*, volume 12170 of *LNCS*, pages 339–368. Springer, Cham, August 2020.
9. Sonia Belaïd, Darius Mercadier, Matthieu Rivain, and Abdul Rahman Taleb. IronMask: Versatile verification of masking security. In *2022 IEEE Symposium on Security and Privacy*, pages 142–160. IEEE Computer Society Press, May 2022.
10. Sonia Belaïd, Victor Normand, and Matthieu Rivain. Masked circuit compiler in the cardinal random probing composability framework. In Goichiro Hanaoka and Bo-Yin Yang, editors, *Advances in Cryptology - ASIACRYPT 2025 - 31st International Conference on the Theory and Application of Cryptology and Information Security, Melbourne, VIC, Australia, December 8-12, 2025, Proceedings, Part II*, Lecture Notes in Computer Science, pages 375–406. Springer, 2025.
11. Sonia Belaïd, Matthieu Rivain, and Mélissa Rossi. New techniques for random probing security and application to raccoon signature scheme. In Serge Fehr and Pierre-Alain Fouque, editors, *EUROCRYPT 2025, Part VIII*, volume 15608 of *LNCS*, pages 94–123. Springer, Cham, May 2025.
12. Sonia Belaïd, Matthieu Rivain, and Abdul Rahman Taleb. On the power of expansion: More efficient constructions in the random probing model. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part II*, volume 12697 of *LNCS*, pages 313–343. Springer, Cham, October 2021.
13. Sonia Belaïd, Matthieu Rivain, Abdul Rahman Taleb, and Damien Vergnaud. Dynamic random probing expansion with quasi linear asymptotic complexity. In Mehdi Tibouchi and Huaxiong Wang, editors, *ASIACRYPT 2021, Part II*, volume 13091 of *LNCS*, pages 157–188. Springer, Cham, December 2021.
14. Francesco Berti, Sebastian Faust, and Maximilian Orlt. Provable secure parallel gadgets. *IACR TCHES*, 2023(4):420–459, 2023.
15. Gaëtan Cassiers, Sebastian Faust, Maximilian Orlt, and François-Xavier Standaert. Towards tight random probing security. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part III*, volume 12827 of *LNCS*, pages 185–214, Virtual Event, August 2021. Springer, Cham.
16. Suresh Chari, Charanjit S. Jutla, Josyula R. Rao, and Pankaj Rohatgi. Towards sound approaches to counteract power-analysis attacks. In Michael J. Wiener, editor, *CRYPTO'99*, volume 1666 of *LNCS*, pages 398–412. Springer, Berlin, Heidelberg, August 1999.
17. Jean-Sébastien Coron and Lorenzo Spignoli. Secure wire shuffling in the probing model. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part III*, volume 12827 of *LNCS*, pages 215–244, Virtual Event, August 2021. Springer, Cham.
18. Alexandre Duc, Stefan Dziembowski, and Sebastian Faust. Unifying leakage models: From probing attacks to noisy leakage. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 423–440. Springer, Berlin, Heidelberg, May 2014.
19. Stefan Dziembowski, Sebastian Faust, and Karol Zebrowski. Simple refreshing in the noisy leakage model. In Steven D. Galbraith and Shiho Moriai, editors, *ASIACRYPT 2019, Part III*, volume 11923 of *LNCS*, pages 315–344. Springer, Cham, December 2019.

20. Louis Goubin and Jacques Patarin. DES and differential power analysis (the "duplication" method). In Çetin Kaya Koç and Christof Paar, editors, *CHES'99*, volume 1717 of *LNCS*, pages 158–172. Springer, Berlin, Heidelberg, August 1999.
21. Dahmun Goudarzi, Antoine Joux, and Matthieu Rivain. How to securely compute with noisy leakage in quasilinear complexity. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part II*, volume 11273 of *LNCS*, pages 547–574. Springer, Cham, December 2018.
22. Yuval Ishai, Amit Sahai, and David Wagner. Private circuits: Securing hardware against probing attacks. In Dan Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 463–481. Springer, Berlin, Heidelberg, August 2003.
23. Vahid Jahandideh, Bart Mennink, and Lejla Batina. An algebraic approach for evaluating random probing security with application to AES. *IACR TCHES*, 2024(4):657–689, 2024.
24. Vahid Jahandideh, Bart Mennink, and Lejla Batina. Leakage-diagrams, importance sampling, and composition in the random probing model. Cryptology ePrint Archive, Paper 2026/444, 2026.
25. Emmanuel Prouff and Matthieu Rivain. Masking against side-channel attacks: A formal security proof. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 142–159. Springer, Berlin, Heidelberg, May 2013.
26. Weijia Wang, Pierrick Méaux, Gaëtan Cassiers, and François-Xavier Standaert. Efficient and private computations with code-based masking. *IACR TCHES*, 2020(2):128–171, 2020.

A Proof of Lemma 1

Proof. By definition of secret independence, for every leakage set $\mathcal{W}$ such that C is secret independent for $\mathcal{W}$, there exists a simulator perfectly simulating the leakage distribution for $\mathcal{W}$.

Now consider $\mathcal{W} \leftarrow \mathsf{LeakingWires}(C, p)$. By assumption, with probability at least $1 - \varepsilon$, the sampled leakage set $\mathcal{W}$ is such that C is secret independent. Hence the simulator fails only with probability at most ε. Hence there exists a simulator whose output distribution differs from the real random probing leakage distribution by statistical distance at most ε. Therefore, C is (p, ε)-random probing secure. □

B Proofs for Section 4

B.1 Proofs for Section 4.1

We first introduce a lemma that characterizes which sets of intervals lead to partial sums that are jointly independent of the masked value.

Lemma 9 (Secret independence of partial sums with incomplete intervals). *Let $[\![x]\!] = \mathsf{Enc}(x)$ be a uniform additive encoding of $x \in \mathbb{K}$, and let $\mathcal{S}$ be a finite set of share intervals. Then*

$$\overline{\mathcal{S}} \cap \Omega = \emptyset \qquad \textit{(i.e. the closure of } \mathcal{S} \textit{ does not contain any complete interval)}$$

if and only if the distribution of $[\![x]\!]_{\mathcal{S}}$ does not depend on x.

Proof. Let us first prove the right-to-left implication. Taking the contraposite, we can assume that there exists $(i, i + kn - 1) \in \overline{\mathcal{S}}$ for some $k > 0$. It follows that $[\![x]\!]_{(i,i+kn-1)} = kx$, which depends on x, can be written as a linear combination of $[\![x]\!]_{\mathcal{S}}$, which therefore also depends on x.

Turning to the left-to-right implication, let us first define $\mathcal{S}'$ by the following process: initialize $\mathcal{S}' \leftarrow \mathcal{S}$. Then, iteratively apply the following update rules: (i) if (i, j) and (i, k) belong to $\mathcal{S}'$ with $j < k$, then remove (i, k) from $\mathcal{S}'$ and add $(j + 1, k)$ to it, and (ii) if (i, k) and (j, k) belong to $\mathcal{S}'$ with $i < j$, then remove (i, k) from $\mathcal{S}'$ and add $(i, j - 1)$ to it. This process converges since each update strictly reduces the sum of the sizes of the share intervals in $\mathcal{S}'$. Further, in the resulting $\mathcal{S}'$, there are are no intervals that have identical first points, nor intervals with identical last points. We also remark that the update operations do not change the closure of the set, hence $\overline{\mathcal{S}'} = \overline{\mathcal{S}}$.

Let us now enumerate $\mathcal{S}'$. We denote $\sigma = |\mathcal{S}'|$, and (s_i, t_i) such that $\mathcal{S}' = \{(s_i, t_i) \mid i \in [\sigma]\}$. Let us further assume that we work in a field of characteristic q and let $A \in [q]^{\sigma \times n}$, $A = (a_{i,j})_{i,j}$ with

$$a_{i,j} = |\{k \in [s_i, t_i] : k \bmod n = j\}| \bmod q.$$

This construction mean that the partial sums of $[\![x]\!]$ associated to $\mathcal{S}'$ can be computed as $A[\![x]\!]$.

By contradiction, let us assume that $[\![x]\!]_{\overline{\mathcal{S}}} = [\![x]\!]_{\overline{\mathcal{S}'}}$ is not independent of x. Since $[\![x]\!]_{\overline{\mathcal{S}'}}$ can be derived from $[\![x]\!]_{\mathcal{S}'}$, the latter is also not independent of x. By [26, Lemma 1], the distribution $[\![x]\!]_{\mathcal{S}'}$ is characterized by the set of all linear combinations of its elements. This means that that there exists $\beta \in [q]$ such that $\beta^T A[\![x]\!]$ depends on x. This implies that there exists $\gamma \in [1, q]$ such that $\beta^T A = (\gamma, \ldots, \gamma)$ (we prove this by decomposing the inner produce $(\beta^T A)[\![x]\!]$ into the sum of a multiple of x and a linear combination of $[\![x]\!]_{[n-1]}$, then remark that this linear combination must have all-0 coefficients to have dependency on x).

For any $j \in [n]$, there is at most one share interval in $\mathcal{S}'$ whose end is congruent to $j \pmod n$, and at most one that whose first element is congruent to $j+1 \pmod n$, thanks to the algorithm that simplifies $\mathcal{S}'$. Let us further remark that if (s_i, t_i) is complete, then $a_{i,j} = a_{i,j'}$ for all j and j'. If this share interval is incomplete, then $a_{i,t_i} - a_{i,t_i+1} = 1$, $a_{i,s_i-1} - a_{i,s_i} = -1$ and $a_{i,j} - a_{i,j+1} = 0$ for all $j \in [n] \setminus \{s_i - 1, t_i\}$. Then, if an interval $(s_i, t_i) \in \mathcal{S}'$ is such that $t_i = j \mod n$ and $\beta_i \neq 0$, there must exist an interval $(s_{i'}, t_{i'}) \in \mathcal{S}'$ with $s_{i'} = j+1 \mod n$ and $\beta_{i'} = \beta_i$. Indeed, if (s_i, t_i) is incomplete (otherwise, $i' = i$ is a solution), we have $a_{i,j} - a_{i,j+1} = 1$, there must be at least one $i' \neq i$ such that $a_{i',j} \neq a_{i',j+1}$. Since $(s_{i'}, t_{i'})$ end cannot be $j \pmod n$ (by unicity of share interval end index and $i' \neq i$), we have $s_{i'} = j+1 \pmod n$. Further, there is a single i' that satisfies the required conditions, which implies $\beta_i = \beta_{i'}$.

We conclude by remarking that there is at least one share interval $(s_{i_1}, t_{i_1}) \in \mathcal{S}'$ such that $\beta_{i_1} \neq 0$. The previous argument then yields i_2 such that $\beta_{i_2} = \beta_{i_1} \neq 0$. Iteratively following this argument, we necessarily end up repeating one index i_*. In the corresponding sequence of share interals, starting from i_* Let I be the finite sequence of share intervals in $\mathcal{S}'$, starting with i_* and terminating before it repeats. The share intervals in I can be concatenated, and their concatenation is complete. □

Next we state and prove the composation properties of simulatability.

Proposition 10 (Simulatability of sequential composition). *Let G_1 be a ℓ_1-to-ℓ_2 gadget and G_2 be a ℓ_2-to-ℓ_3 gadget. Let $\mathcal{W}_1$ and $\mathcal{W}_2$ be sets of wires in G_1 and G_2, and $\boldsymbol{I_1}$, $\boldsymbol{O_1}$ and $\boldsymbol{O_2}$ be respectively a ℓ_1-, ℓ_2- and ℓ_3-tuple of sets of share intervals. If G_1 is $(\boldsymbol{I_1}, \boldsymbol{O_1}, \mathcal{W}_1)$-simulatable and G_2 is $(\boldsymbol{O_1}, \boldsymbol{O_2}, \mathcal{W}_2)$-simulatable, then $G_2 \circ G_1$ (represented on Figure 4a) is $(\boldsymbol{I_1}, \boldsymbol{O_2}, \mathcal{W}_1 \cup \mathcal{W}_2)$-simulatable.*

Proof. By assumption, G_1 is $(\boldsymbol{I_1}, \boldsymbol{O_1}, \mathcal{W}_1)$-simulatable and G_2 is $(\boldsymbol{O_1}, \boldsymbol{O_2}, \mathcal{W}_2)$-simulatable. Let Sim_1 and Sim_2 be their respective simulators.

We define a simulator Sim for the sequential composition $G_2 \circ G_1$ as follows. On input $[\![\boldsymbol{x}]\!]_{\boldsymbol{I_1}}$, the simulator first runs Sim_1 to obtain values distributed like the output of $\mathsf{AssignWires}\,(G_1, \mathcal{W}_1 \cup [\![\boldsymbol{y}]\!]_{\boldsymbol{O_1}}, [\![\boldsymbol{x}]\!])$ and then feeds those values corresponding to the share $[\![\boldsymbol{y}]\!]_{\boldsymbol{O_1}}$ to Sim_2 in order to obtain values distributed as $\mathsf{AssignWires}\,(G_2, \mathcal{W}_2 \cup [\![\boldsymbol{z}]\!]_{\boldsymbol{O_2}}, [\![\boldsymbol{y}]\!])$. Finally, Sim outputs the values it received from Sim_1 and Sim_2, identically distributed to

$$\mathsf{AssignWires}\,(G_2 \circ G_1, \mathcal{W}_1 \cup \mathcal{W}_2 \cup [\![\boldsymbol{z}]\!]_{\boldsymbol{O_2}}, [\![\boldsymbol{x}]\!])\,.$$

Therefore, $G_2 \circ G_1$ is $(\boldsymbol{I_1}, \boldsymbol{O_2}, \mathcal{W}_1 \cup \mathcal{W}_2)$-simulatable. □

Proposition 11 (Simulatability of parallel composition). *Let G_1 be a ℓ_1-to-ℓ_2 gadget and G_2 be a ℓ_3-to-ℓ_4 gadget. Let $\mathcal{W}_1$ and $\mathcal{W}_2$ be sets of wires in G_1 and G_2, let $\boldsymbol{I_1}$, $\boldsymbol{O_1}$, $\boldsymbol{I_2}$, $\boldsymbol{O_2}$ be respectively a ℓ_1-, ℓ_2-, ℓ_3- and ℓ_4-tuple of sets of share intervals. If G_1 is $(\boldsymbol{I_1}, \boldsymbol{O_1}, \mathcal{W}_1)$-simulatable and G_2 is $(\boldsymbol{I_2}, \boldsymbol{O_2}, \mathcal{W}_2)$-simulatable, then $G_1 \| G_2$ (represented on Figure 4b) is $((\boldsymbol{I_1}, \boldsymbol{I_2}), (\boldsymbol{O_1}, \boldsymbol{O_2}), \mathcal{W}_1 \cup \mathcal{W}_2)$-simulatable.*

Proof. By assumption, G_1 is $(\boldsymbol{I_1}, \boldsymbol{O_1}, \mathcal{W}_1)$-simulatable and G_2 is $(\boldsymbol{I_2}, \boldsymbol{O_2}, \mathcal{W}_2)$-simulatable. Let Sim_1 and Sim_2 be their respective simulators.

We define a simulator Sim for the parallel composition $G_1 \| G_2$ as follows. On input $([\![\boldsymbol{w}]\!]_{\boldsymbol{I_1}}, [\![\boldsymbol{x}]\!]_{\boldsymbol{I_2}})$, the simulator first runs Sim_1 on $[\![\boldsymbol{w}]\!]_{\boldsymbol{I_1}}$ to obtain a pair distributed as $\mathsf{AssignWires}(G_1, \mathcal{W}_1 \cup [\![\boldsymbol{y}]\!]_{\boldsymbol{O_1}}, [\![\boldsymbol{w}]\!])$, and then runs Sim_2 on $[\![\boldsymbol{x}]\!]_{\boldsymbol{I_2}}$ to obtain a pair distributed as $\mathsf{AssignWires}(G_2, \mathcal{W}_2 \cup [\![\boldsymbol{y}]\!]_{\boldsymbol{O_2}}, [\![\boldsymbol{w}]\!])$. Finally, Sim outputs the results from the two simulators. Since the gadgets have independent randomness, the distribution of the simulator output is identically distributed to

$$\mathsf{AssignWires}\,(G_1 \| G_2, \mathcal{W}_1 \cup \mathcal{W}_2 \cup [\![\boldsymbol{y}]\!]_{\boldsymbol{O_1}} \cup [\![\boldsymbol{z}]\!]_{\boldsymbol{O_2}}, [\![\boldsymbol{w}]\!], [\![\boldsymbol{x}]\!])\,,$$

therefore $G_1 \| G_2$ is $((\boldsymbol{I_1}, \boldsymbol{I_2}), (\boldsymbol{O_1}, \boldsymbol{O_2}), \mathcal{W}_1 \cup \mathcal{W}_2)$-simulatable. □

Proof of Proposition 1

Proof. Let $\boldsymbol{I} = (\overline{I}_1(\mathcal{W}), \ldots, \overline{I}_{\ell_1}(\mathcal{W}))$ and let $[\![\boldsymbol{x}]\!]$ the input sharings. It suffices to show that if for all i, $\overline{I}_i(\mathcal{W}) \cap \Omega = \emptyset$, then $[\![\boldsymbol{x}]\!]_{\boldsymbol{I}}$ is independent of the unmasked secret (where $[\![\boldsymbol{x}]\!]$ is made of independent uniform sharings). For each input sharing $[\![x]\!]^i$, if $\overline{I}_i(\mathcal{W}) \cap \Omega = \emptyset$ and $[\![x]\!]^i$ a uniform sharing of x^i, then $[\![x]\!]^i_{I_i(\mathcal{W})}$ is independent of x^i by Lemma 9. □

Proof of Proposition 2

Proof. For each $i \in [n]$, if G_i contains a wire in $\mathcal{W}$, then we include i in I. Let $I \subset [n]$ be such that $i \in I$ iff a wire of G_i belongs to $\mathcal{W}$. Simulating $\mathcal{W}$ using $\boldsymbol{x}_I$ is then trivial, and $|I| \leq |\mathcal{W}|$ since the G_i are disjoint. Considering any share interval (i, j), the affine property gives $[\![\boldsymbol{y}]\!]_{(i,j)} = l_g([\![\boldsymbol{x}]\!]_{(i,j)}) + \sum_{k=i}^{j} c_k$ (where $c_k = g(0) - l_g(0)$ if $k = 0 \mod n$ and $c_k = 0$ otherwise), showing that the output partial sum can be simulated using the input partial sum with the same interval. □

Proof of Lemma 2

Proof. The ORS simulator can trivially simulate the values of the wires $\mathcal{W}$ from the generalized free-SNI simulator, along with the values of the output wires indexed by I.

Next, given a set of share intervals O, let us assume that $I' \cup O$ is complete. The ORS simulator knows the unmasked value of the inputs, hence it can compute the one for the output, and thus simulate all the output shares ($[\![\boldsymbol{y}]\!]_I$ from the generalized free-SNI simulator, and $[\![\boldsymbol{y}]\!]_{[n]\setminus I}$ as uniform sharings under the condition that $[\![\boldsymbol{y}]\!]$ sums to the correct value).

In the case where $I' \cup O$ is incomplete, the outputs $[\![\boldsymbol{y}]\!]_{I' \cup O}$ are independent on the unmasked outputs by Lemma 9. We can therefore perform the simulation for an arbitrary $\boldsymbol{y}$ (which is a possible output of the gadget, *i.e.*, that corresponds to some input $\boldsymbol{x}$). We then complete the simulation as in the previous case. □

B.2 Proofs for Section 4.2

Coverings are closed

Lemma 10 (Covering is closed). *For any covering* $\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}$,

$$\overline{\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}} = \mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}.$$

Proof. If $\Gamma^{\mathcal{W},\boldsymbol{O}}_{C,\Phi}$ contains an orbit, $\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i} = \Omega^*$, and $\overline{\Omega^*} = \Omega^*$. Otherwise, each share interval in $\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}$ corresponds to a walk in $\Gamma^{\mathcal{W},\boldsymbol{O}}_{C,\Phi}$. The addition and subtraction of such share intervals corresponds to a concatenation of their corresponding walks such that the concatenated walk corresponds to the sum/difference of the share intervals. Therefore, any ω in $\overline{\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}}$ also belongs to $\mathcal{C}^{\mathcal{W},\boldsymbol{O}}_{C_n,i}$. □

Composition of coverings

Proposition 12 (Covering Sequential Composition). *Let C_1 be a ℓ_1-to-ℓ_2 gadget and C_2 be a ℓ_2-to-ℓ_3 gadget, and let $\mathcal{W}_1$ and $\mathcal{W}_2$ be sets of leaking wires in C_1 and C_2, respectively. Then, for any ℓ_3-tuple of sets of share intervals $\boldsymbol{O}$,*

$$\mathcal{C}^{\mathcal{W}_1 \cup \mathcal{W}_2,\boldsymbol{O}}_{C} \supset \mathcal{C}^{\mathcal{W}_1,\boldsymbol{I_2}}_{C_1} \qquad \textit{where} \qquad \boldsymbol{I_2} = \mathcal{C}^{\mathcal{W}_2,\boldsymbol{O}}_{C_2},$$

where the set inclusion is element-wise on the tuple.

Proof. Let us pose $C = C_2 \circ C_1$, $\mathcal{W} = \mathcal{W}_1 \cup \mathcal{W}_2$, and denote by $[\![\boldsymbol{y}]\!]$ the tuples of output sharings of C_1, which are also the input sharings of C_2.

Let us first assume that $\Gamma_{C_2,\Phi}^{\mathcal{W}_2,\boldsymbol{O}}$ contains an orbit. This implies that $\boldsymbol{I_2} = \Omega^*$, which means that $\mathcal{C}_C^{\mathcal{W}_1\cup\mathcal{W}_2,\boldsymbol{O}} = \mathcal{C}_{C_1}^{\mathcal{W}_1,\boldsymbol{I_2}} = \Omega^*$.

Otherwise, let us remark that any edge $\{\eta,\eta'\}$ in $\Gamma_{C_1,\Phi}^{\mathcal{W}_1,\boldsymbol{I_2}}$ has an equivalent walk in $\Gamma_{C,\Phi}^{\mathcal{W},\boldsymbol{O}}$ whose extremities are a and b. This walk exists in $\Gamma_{C_2,\Phi}^{\mathcal{W}_2,\boldsymbol{O}}$ by the definition of covering on C_2 for edges of the form $\nu_{(a,b)}^{[\![y^i]\!]}$ (for some a and b), and this walk also exists in $\mathcal{C}_C^{\mathcal{W}_1\cup\mathcal{W}_2,\boldsymbol{O}}$ since and any edge of $\Gamma_{C_2,\Phi}^{\mathcal{W}_2,\boldsymbol{O}}$ also belongs to $\mathcal{C}_C^{\mathcal{W}_1\cup\mathcal{W}_2,\boldsymbol{O}}$. Any other edge of $\Gamma_{C_1,\Phi}^{\mathcal{W}_1,\boldsymbol{I_2}}$ also belongs to $\Gamma_{C,\Phi}^{\mathcal{W},\boldsymbol{O}}$ and is thus itself a walk from a to b. Therefore, for any walk in $\Gamma_{C_1,\Phi}^{\mathcal{W}_1,\boldsymbol{I_2}}$, there is an equivalent walk in $\Gamma_{C,\Phi}^{\mathcal{W},\boldsymbol{O}}$ with the same extremities (we obtain it by replacing each edge by the walk corresponding to that edge).

Finally, we can conclude by remarking that if there is an orbit in $\Gamma_{C_1,\Phi}^{\mathcal{W}_1,\boldsymbol{I_2}}$ then the equivalent walk in $\Gamma_{C,\Phi}^{\mathcal{W},\boldsymbol{O}}$ is also an orbit, and $\mathcal{C}_C^{\mathcal{W}_1\cup\mathcal{W}_2,\boldsymbol{O}} = \mathcal{C}_{C_1}^{\mathcal{W}_1,\boldsymbol{I_2}} = \Omega^*$. Otherwise, any walk equivalent to an input arc in $\Gamma_{C_1,\Phi}^{\mathcal{W}_1,\boldsymbol{I_2}}$ has its equivalent walk in $\Gamma_{C,\Phi}^{\mathcal{W},\boldsymbol{O}}$ which is equivalent to the same input arc. This implies that $\mathcal{C}_{C_1}^{\mathcal{W}_1,\boldsymbol{I_2}} \subset \mathcal{C}_C^{\mathcal{W}_1\cup\mathcal{W}_2,\boldsymbol{O}}$. □

Proposition 13 (Covering parallel composition). *Let C_1 be a ℓ_1-to-ℓ_2 gadget and C_2 be a ℓ_3-to-ℓ_4 gadget, and let let $\mathcal{W}_1$ and $\mathcal{W}_2$ be sets of leaking wires in C_1 and C_2. Then, for any ℓ_2-tuple and ℓ_4-tuple of sets of share intervals $\boldsymbol{O}$ and $\boldsymbol{O'}$,*

$$\mathcal{C}_{C_1\|C_2}^{\mathcal{W}_1\cup\mathcal{W}_2,(\boldsymbol{O},\boldsymbol{O'})} \supset \left(\mathcal{C}_{C_1}^{\mathcal{W}_1,\boldsymbol{O}}, \mathcal{C}_{C_2}^{\mathcal{W}_2,\boldsymbol{O'}}\right),$$

where the set inclusion is element-wise on the tuples.

Proof. Since the extended leakage diagram $\Gamma_{C_1\|C_2,\Phi}^{\mathcal{W}_1\cup\mathcal{W}_2,(\boldsymbol{O},\boldsymbol{O'})}$ is the union of the disconnected subgraphs $\Gamma_{C_1,\Phi}^{\mathcal{W}_1,\boldsymbol{O}}$ and $\Gamma_{C_2,\Phi}^{\mathcal{W}_2,\boldsymbol{O'}}$, the sets of paths equivalent to input arcs are preserved, and in particular the presence of an orbit. Let us remark that the equality is only lost when exactly one of the two graphs contains an orbit. □

B.3 Proofs for Section 4.3

Proof of Proposition 5

Proof. Let R be a circular refresh gadget. We recall that it is a 1-to-1 gadget. Let O be a set of output share intervals, and let $\mathcal{W}$ be a set of internal wires. We have to prove that R is $(\mathcal{C}_{\mathcal{W},O}^R, O, \mathcal{W})$-simulatable.

The extended leakage diagram of R contains the input nodes $\eta_i^{[\![x]\!]}$ and output nodes $\eta_i^{[\![y]\!]}$ for $i \in [n]$, and the following edges, where each edge corresponds to a probe and takes its value:

- $V\left(\left(\eta_{i+1 \bmod n}^{[\![x]\!]}, \eta_{i+1 \bmod n}^{[\![y]\!]}\right)\right) = r_i$,
- $V\left(\left(\eta_i^{[\![x]\!]}, \eta_{i+1 \bmod n}^{[\![y]\!]}\right)\right) = b_i$,
- $V\left(\left(\eta_i^{[\![x]\!]}, \eta_{i+1 \bmod n}^{[\![x]\!]}\right)\right) = x_i$,
- $V\left(\left(\eta_i^{[\![y]\!]}, \eta_{i+1 \bmod n}^{[\![y]\!]}\right)\right) = y_i$,
- $V\left(\left(\eta_i^{[\![y]\!]}, \eta_{j+1 \bmod n}^{[\![y]\!]}\right)\right) = [\![y]\!]_{(i,j)}$ (coming from any $(i,j) \in O$).

In Figure 8, we give an example of extended leakage diagram.

For the sake of this proof, in the extended leakage diagram of R, we consider an *orientation* for the edges: we denote each edge as a couple of nodes (η,η'), and assign a *value* to every edge, $V((\eta,\eta'))$, such that changing the orientation of an edge changes the sign of its value: $V((\eta',\eta)) = -V((\eta,\eta'))$. We can

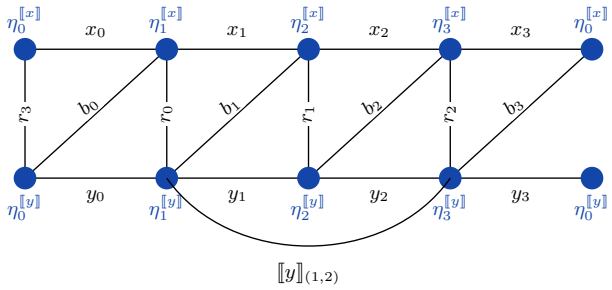


Fig. 8: Example of extended leakage diagram for a circular refresh gadget with $n = 4$ and $O = \{(1, 2)\}$.

thus give a value to a walk by summing the values of its edges: for a walk ω whose associated vertices are $(v_0, \ldots, v_k)$, the value is

$$V(\omega) = \sum_{i=0}^{k-1} V((v_i, v_{i+1})).$$

We make two observations that will be useful for the proof.

Observation 1. Let us first remark that walk equivalence preserves the value of walks. Indeed, all equivalences given in Definition 8 relate walks with the same value (assuming the sub-walks have the same values), as well as the equivalence defined in Definition 15. Therefore, any *closed* walk (a cycle) which is not an orbit has value 0. Indeed, through elementary equivalences, any walk can be transformed to contain only $[\![x]\!]_i$ and r_i edges, and r_i edges can the be removed: any r_i in the walk appears twice consecutively, or it is both the first and last edge of the walk. Then, each edge $[\![x]\!]_i$ repeated twice consecutively in the walk can be removed, leaving an empty walk (that sums to 0) if it is not an orbit.

Observation 2. As a second remark, let us consider a node $\eta_{i+1 \bmod n}^{[\![y]\!]}$. All incident edges of this node are r_i, b_i, y_i, $y_{i-1 \bmod n}$ and any edge coming from share intervals in O of the form (k, i) or $(i + 1, k)$. We will show that all these nodes contain the random value r_i as an additive term in their expression, and that no other edge contains r_i as an additive term. Therefore, if there is a single edge incident to $\eta_{i+1 \bmod n}^{[\![y]\!]}$ in the extended leakage diagram, then we can simulate its value as a fresh random value, independently of all other values to simulate of the input shares. Let us now prove that r_i appears as an additive term in the expression of all these edges. It appears in the edge labelled by r_i by definition. It also appears in the edge labelled by b_i since $b_i = r_i + x_i$. It also appears in the edge labelled by y_i since $y_i = r_i + x_i - r_{i-1}$, and thus also in the edge labelled by $y_{i-1 \bmod n} = r_{i-1 \bmod n} + x_{i-1 \bmod n} - r_i$. Finally, it appears in any edge coming from share intervals in O that contain i. Let us assume a share interval (k, i), $[\![y]\!]_{(k,i)} = \sum_{j=k}^{i} y_j = \sum_{j=k}^{i}(r_j + x_j - r_{j-1}) = \left(\sum_{j=k}^{i} x_j\right) + r_i - r_{k-1}$, and thus contains r_i as an additive term. The case of share intervals $(i + 1, k)$ is similar. In addition, one can check that no other edge contains r_i as an additive term.

Let us now go back to the main proof. If there is an orbit in the extended leakage diagram, the simulation is trivial, since the simulator has access to all the input shares. Otherwise, we proceed by induction on the size of the set of probes (i.e., edges in the extended leakage diagram), showing that we can remove probes without loss of generality (i.e., if we can simulate a smaller set, then we can simulate the larger one), where the base case (the empty set) is trivial. Given an extended leakage diagram, apply the following algorithm for finding probes to remove (we show how to simulate the probe from the simulation of the other probes and the input partial sums corresponding to the covering).

1. If there is a cycle in the extended leakage diagram (other than an orbit): its value is 0 (by Observation 1), hence we can simulate the value of the first edge as the opposite of the value of the remaining part of the walk.

2. Otherwise, if there is a path equivalent to an input arc $\alpha_{(i,j)}^{[\![x]\!]}$, its value is equal to $[\![x]\!]_{(i,j)}$ (by Observation 1), which the simulator knows since (i,j) belongs to the covering. Therefore, we can again simulate the value of the first edge of the path from the value of the remaining part of the path (and from $[\![x]\!]_{(i,j)}$).
3. Otherwise, as a result of negating the previous condition 1, the extended leakage diagram is a union of acyclic components, i.e., a forest (a union of disconnected trees). Moreover, the negation of condition 2 means that at most one leaf (i.e. extremity) of each component belongs to $\{\eta_i^{[\![y]\!]}\}_{i\in[n]}$ (otherwise, the path between the two leaves would be equivalent to an arc). Since a tree with at least one edge contains at least two leaves, there is at least one leaf $\eta_i^{[\![y]\!]}$ in the leakage diagram, which has one incident edge. We can thus simulate that edge as a fresh random (by Observation 2).

□

Proof of Proposition 6

Proof. Considering any set of wires $\mathcal{W}$ in C and tuple of sets of output share intervals $\boldsymbol{O}$, let us show that we can simulate using the covering. If there is an orbit, simulation is trivial since the covering contains all the input shares. Otherwise, since the dependency graph is only a single circle, the covering is $I' \cup \bigcup_i O_i$, which is the set of share intervals needed by the ORS simulator to simualte $\mathcal{W}$ along with all output share subsets corresponding to the intervals in $\bigcup_i O_i$. □

Proof of composition for covering simulatability (Proposition 4)

Lemma 11 (Covering Simulatability for Sequential Composition). *If the gadgets C_1 and C_2 are simulatable by their coverings, then their sequential composition $C_2 \circ C_1$ is simulatable by its covering.*

Proof. Let $\mathcal{W}_1$ and $\mathcal{W}_2$ be sets of wires in C_1 and C_2. Thanks to simulatability by covering, we know that for any $\boldsymbol{O_2}$, C_2 is $(\boldsymbol{I_2}, \boldsymbol{O_2}, \mathcal{W}_2)$-simulatable with $\boldsymbol{I_2} = \mathcal{C}_{C_2}^{\mathcal{W}_2, \boldsymbol{O_2}}$, and C_1 is $(\boldsymbol{I_1}, \boldsymbol{I_2}, \mathcal{W}_1)$-simulatable with $\boldsymbol{I_1} = \mathcal{C}_{C_1}^{\mathcal{W}_1, \boldsymbol{I_2}}$. Therefore, by Proposition 10, $C_2 \circ C_1$ is $(\boldsymbol{I_1}, \boldsymbol{O_2}, \mathcal{W}_1 \cup \mathcal{W}_2)$-simulatable. By Proposition 12, we have $\mathcal{C}_{C_2 \circ C_1}^{\mathcal{W}_1 \cup \mathcal{W}_2, \boldsymbol{O_2}} \supset \boldsymbol{I_1}$, which shows that it is sufficient for the simulation. □

Lemma 12 (Covering Simulatability for Parallel Composition). *If the gadgets C_1 and C_2 are simulatable by their coverings, then their parallel composition $C_1 \| C_2$ is simulatable by its covering.*

Proof. Similarly to the proof of Lemma 11, this is a direct consequence of Proposition 13 and Proposition 11. □

Let us now prove Proposition 4.

Proof. Let the circuit C be composed of the gadgets be $C_1, \ldots, C_\kappa$ (sorted in topological order). Extend each gadget C_i' by composing it in parallel with any number of identity gadgets, and sequentially compose it with a gadget C_i'' that re-orders sharings, such that $C = (C_\kappa'' \circ C_\kappa') \circ \cdots \circ (C_1'' \circ C_1')$ (all the wires in identity and re-ordering gadgets disappear in the composition). We then have C written as a sequential composition of covering-simulatable gadgets: C_i'' and C_i' are covering-simulatable by Lemma 12 (the property is trivial for C_i''), therefore we can apply Lemma 11 to C. □

B.4 Proofs for Section 4.4

We first prove the equidistribution of the random probing leakage diagram and the leakage diagram with leakage sampled from LeakingWires.

Proposition 14 (Equivalence between Leakage Diagrams). *Let $n \in \mathbb{N}$, $p \in [0,1]$, C be a circuit, C_n its compiled version, and let Γ_{C_n} be its dependency graph. Let Φ be a mapping assigning to any subset of wires of a multiplication gadget G a subset of indices in $[n]$ such that for all set of wires $\mathcal{W}$, G is $(\Phi(\mathcal{W}), \mathcal{W})$-generalized free SNI.*

When $\mathcal{W} \leftarrow \mathsf{LeakingWires}(C_n, p)$, the leakage diagram $\Gamma_{C_n,\Phi}^{\mathcal{W}}$ of Definition 7 and the random probing leakage diagram $\Gamma_{C_n}^{p}$ of Definition 18 have the same distribution.

Proof. We compare the two constructions gadget by gadget. The set of edges associated to different gadgets are jointly independent in both graph constructions.

For refresh gadgets, an r-edge is present exactly when at least one of its three associated wires leaks. Since each wire leaks independently with probability p, this occurs with probability $1-(1-p)^3$. Similarly, a b-edge corresponds to a single wire and is therefore present with probability p. All edges are present independently of each other, and the edges on the input/output wires are handled by the adjacent non-refresh gadgets.

For affine gadgets, an edge ν_i^G is present exactly when at least one of the α input/output wires associated with share index i leaks. Hence it appears with probability $1-(1-p)^\alpha$, independently of the other edges.

Finally, consider a multiplication gadget G. In the leakage diagram of Definition 7, the set of edges added for G is $(\nu_i^G)_{i\in\Phi(\mathcal{W}_G)}$, where $\mathcal{W}_G$ is the restriction of $\mathcal{W}$ to the wires of G. Therefore, for any $I\subset[n]$, this set is equal to $(\nu_i^G)_{i\in I}$ exactly when $\Phi(\mathcal{W}_G)=I$. The probability of adding this set of edges is

$$\Pr[\Phi(\mathcal{W}_G)=I \mid \mathcal{W}_G \leftarrow \mathsf{LeakingWires}(G,p)]\,,$$

which is precisely the rule of Definition 18. □

Proof of Corollary 1

Proof. By Theorem 1, if the leakage diagram associated with a leakage set $\mathcal{W}$ contains no orbit, then C_n is secret independent for $\mathcal{W}$.

In the random probing setting, let

$$\varepsilon' = \Pr\big[\Gamma^{\mathcal{W}}_{C_n,\Phi} \text{ contains an orbit}\big],$$

where $\Gamma^{\mathcal{W}}_{C_n,\Phi}$ is the leakage diagram associated with $\mathcal{W}\leftarrow\mathsf{LeakingWires}(C_n,p)$. Then, with probability at least $1-\varepsilon'$, the circuit is secret independent.

By Proposition 14, the distribution of $\Gamma^{\mathcal{W}}_{C_n,\Phi}$ coincides with that of the random probing leakage diagram $\Gamma^p_{C_n}$. Therefore, $\varepsilon=\varepsilon'$.

Hence, with probability at least $1-\varepsilon$, the random probing leakage is perfectly simulatable. The random probing simulator may thus fail only with probability at most ε, which proves that C_n is (p,ε)-random probing secure. □

C Proofs for Section 5

C.1 Proof of Proposition 7

Proof. Let $I=\Phi_{G_{\text{mult}}}(\mathcal{W})$. If $I=[n]$, then the simulator is given all input shares. It can therefore perfectly simulate the full execution of the gadget, including all internal wires in $\mathcal{W}$ and the output sharing, by honestly running Algorithm 2.

Assume now that $I\subsetneq[n]$, and let

$$J=[n]\setminus I.$$

By definition of I, for every $j\in J$ we have

$$T_j\cap\mathcal{W}=\emptyset.$$

We show that the leakage on $\mathcal{W}$ and the output shares indexed by I can be simulated from the input shares $[\![a]\!]_I$ and $[\![b]\!]_I$ only, and that the remaining output shares form a uniform sharing independent from the other ones and the leakage.

First, consider the wires in $\mathcal{W}$. Any wire containing a variable of the form a_i, b_i, or $a_i\cdot b_j$ belongs to the set T_i or T_j with the corresponding index in I. Hence all the input shares needed to compute such a wire are available to the simulator. The same holds for wires of the form $w_{i,j}$. Indeed, if $w_{i,j}\in\mathcal{W}$, then the definition of the sets T_i and T_j implies that both indices involved in this computation belong to I. The simulator can therefore sample the required randomness and compute $w_{i,j}$ exactly as in the real execution,

using the available shares a_i, a_j, b_i, b_j. It remains to consider wires of the form $z_{\ell,i,j}$. If such a wire belongs to $\mathcal{W}$, then the indices i and j are in I. Thus the simulator can compute all terms depending only on available shares, such as products $a_i b_i$. For each intermediate term of the form

$$a_i b_k + z_{0,k,i} + a_k b_i,$$

there are two cases. If some wire depending on the index k is in $\mathcal{W}$, then $k \in I$, and the simulator has access to a_k and b_k, so it can simulate the term honestly. Otherwise, no wire in $\mathcal{W}$ reveals the corresponding randomness $z_{0,k,i}$ or any dependent value involving index k. Since $z_{0,k,i}$ is uniformly random and independent, the whole sum

$$a_i b_k + z_{0,k,i} + a_k b_i$$

is itself uniformly distributed from the simulator's point of view, and can therefore be sampled uniformly.

This gives a perfect simulation of all wires in $\mathcal{W}$ using only $[\![a]\!]_I$ and $[\![b]\!]_I$. The same simulation also produces the output shares $[\![c]\!]_I$, since each such output share is computed from the corresponding wires already simulated.

Since $|J| = n \implies |I| = n$ and by hypothesis $|I| < n$, we have $|J| < n$. Hence, there is at least one output share that is not simulated, we denote it $j^* \in J$. Thus, it remains to justify the simulation of the output shares indexed by $J \setminus \{j^*\}$. For every $j \in J \setminus \{j^*\}$, we have

$$c_j = z_{L,j,0}.$$

Each such output share is a sum of $n-1$ random values. Since no wire in T_j leaked, none of these random value leaks, nor does any intermediate variable containing them; otherwise we would have $T_j \cap \mathcal{W} \neq \emptyset$, contradicting the definition of J. Moreover, each of these random masks appears exactly twice in the computation: once in c_j and once in another output share, such that each pair of output shares has at most one random value in common. Consequently, the only dependency between the shares indexed by J is the global correctness relation

$$\sum_{i=0}^{n-1} c_i = ab.$$

Hence, for any fixed index $j^* \in J$, once the shares $[\![c]\!]_I$ are determined, the collection of output shares

$$(c_j)_{j \in J \setminus \{j^*\}}$$

is uniformly distributed and independent of the leakage and the outputs in I. The remaining share c_{j^*} is then uniquely determined by the correctness condition and is not required to be simulated. □

C.2 Proof of Lemma 3

Proof. Let $i \in [n]$, $L = \log_2(n+1)$. Following the proof of Theorem 4.1 in [14], the $\nu_i^{G_{\text{mult}}}$-*edge* is added to the leakage diagram whenever at least one variable in

$$T_i = \{a_i, b_i, a_i \cdot b_j, a_j \cdot b_i, w_{i,j}, w_{j,i}, z_{\ell,i,j} | j \in [n], \ell \in [L+1]\}$$

leaks its value. We begin by counting the number of wires carrying the value of a variable in T_i, as well as those carrying the value of a variable in $T_i \cap T_j$, for all $j \in [n] \setminus \{i\}$. By enumeration of the different variables in T_i :

1. The variable a_i (resp. b_i) appears in n cross products, requiring n copies of a_i (resp. b_i). By the assumption from [14] on copies, there are $n+1$ wires carrying the value of a_i (resp. b_i).
2. For $j \in [n] \setminus \{i\}$, there are n cross products of the form $a_i \cdot b_j$ and n of the form $a_j \cdot b_i$. Hence, there are $2(n-1)+1$ wires carrying the value of a cross product in T_i (including $a_i \cdot b_i$), among which 2 also belong to T_j.

3. For $j \in [n]\backslash\{i\}$, only $(n-(i+1))$ variables of the form $w_{i,j}$ and i variables of the form $w_{j,i}$ are well defined, each being carried by a single wire. This yields a total of $n-1$ wires carrying the value of such variables in T_i, among which one also belongs to T_j.
4. For $j \in [n]\backslash\{i\}$, if $i < j$, the variable $z_{0,i,j}$ is a random value and thus carried by three wires. On the other hand, if $i > j$, $z_{0,i,j}$ is carried by a single wire. Therefore, there are $3i + (n-i) - 1$ wires carrying variables of this form in T_i. The maximum number of such wires is attained when $i = n-1$ and equals $3(n-1)$. Furthermore, none of these variables belongs to another set T_j (indeed, $z_{0,j,i}$ belongs to T_j but not $z_{0,i,j}$).
5. For each $\ell \in [1, L]$ and each $j \in [0, 2^{L-\ell} - 1]$, the variable $z_{\ell,i,j}$ is well defined and belongs exclusively to T_i. There are

$$\sum_{\ell=1}^{L} \sum_{j=0}^{2^{L-\ell}-1} 1 = \sum_{l=1}^{L} 2^{L-\ell} = 2^L - 1 \leq n$$

wires carrying the value of such variables.

Finally, there are at most

$$2(n+1) + 2n - 1 + (n-1) + 3(n-1) + n = 9n - 3$$

wires carrying the value of a variable belonging to T_i. In addition, for all $j \in [n]\backslash\{i\}$, there are 3 wires that belong to $T_i \cap T_j$ (*i.e.*, $3(n-1)$).

In the following, our proof follows the same path than the proof in [19]. We construct the leakage diagram D as follows :

1. Start with an empty set $D = \{\}$
2. For each $i \in \{0, \ldots, n-1\}$, add T_i to D with probability $1 - (1-p)^{6n}$
3. For each $i, j \in \{0, \ldots, n-1\}$ such that $i \neq j$, add T_i and T_j to D with probability $1 - (1-p)^3$, approximated by $3p$.

The events E_i and E_j, corresponding to adding T_i and T_j to D, are not independent. We modify the leakage diagram to enforce independence between them, as described below.

1. Start with an empty set $D = \{\}$
2. For each $i \in \{0, \ldots, n-1\}$, add T_i to D with probability $1 - (1-p)^{6n}$
3. For each $i, j \in \{0, \ldots, n-1\}$ such that $i \neq j$,
   - Add T_i to D with probability $\sqrt{3p}$
   - Add T_j to D with probability $\sqrt{3p}$

The second construction induces a distribution that dominates that of the first construction and ensures independence between the events E_i and E_j. Consequently, T_i can be added to $\Gamma_{G,\Phi}^{\mathcal{W},p}$ with probability

$$1 - (1-p)^{6n} + 1 - (1 - \sqrt{3p})^{n-1}$$

□

C.3 Proof of Lemma 4

Proof. Let $i, j \in [n]$ such that $i \neq j$. By the proof of Lemma 3, we know that we can add exclusively T_i to $\Gamma_{G,\Phi}^{p}$ with probability $q_1 \triangleq 1 - (1-p)^{6n}$, and T_i with T_j with probability $q_2 \triangleq 1 - (1-p)^3$. Furthermore, a variable of the multiplication is in at most two sets T_i and T_j. We introduce the following events :

- Z_i is set to 1 if the edge T_i is added to the leakage diagram, and to 0 otherwise.

– X_i is set to 1 if the edge T_i is added without adding any other edge T_j, and to 0 otherwise. In this case, $X_i \sim \mathcal{B}(q_1)$.
– $Y_{i,j}$ is set to 1 if both edges T_i and T_j are added to the leakage diagram, and to 0 otherwise. In this case, $Y_{i,j} \sim \mathcal{B}(q_2)$.

Let S be a subset of $[n]$, by the inclusion-exclusion principle, we have

$$\begin{aligned}\Pr\left(\bigwedge_{i\in S} Z_i = 1\right) &= 1 - \Pr\left(\bigvee_{i\in S} Z_i = 0\right)\\ &= 1 - \sum_{\substack{T\subseteq S\\ T\neq\varnothing}} (-1)^{|T|+1} \Pr\left(\bigwedge_{i\in T} Z_i = 0\right)\\ &= 1 + \sum_{\substack{T\subseteq S\\ T\neq\varnothing}} (-1)^{|T|} \Pr\left(\bigwedge_{i\in T} Z_i = 0\right)\end{aligned}$$

Furthermore, when $|T| \geq 1$, we have

$$\begin{aligned}\Pr\left(\bigwedge_{i\in T} Z_i = 0\right) &= \left(\prod_{i\in T} \Pr(X_i = 0)\right)\left(\prod_{\substack{i,j\in T\\ i\neq j}} \Pr(Y_{i,j} = 0)\right)\left(\prod_{k\in[n]\setminus T}\left(\prod_{i\in T} \Pr(Y_{i,k} = 0)\right)\right)\\ &= \left(\prod_{i\in T} \Pr(X_i = 0)\right)\left(\prod_{\substack{i,j\in T\\ i\neq j}} \Pr(Y_{i,j} = 0)\left(\prod_{k\in[n]\setminus T} (1-q_2)^{|T|}\right)\right)\\ &= \left(\prod_{i\in T} \Pr(X_i = 0)\right)\left(\prod_{\substack{i,j\in T\\ i\neq j}} \Pr(Y_{i,j} = 0)(1-q_2)^{|T|(n-|T|)}\right)\\ &= \left(\prod_{i\in T} \Pr(X_i = 0)\right)(1-q_2)^{\frac{|T|(|T|-1)}{2}+|T|(n-|T|)}\\ &= (1-q_1)^{|T|}(1-q_2)^{\frac{|T|(|T|-1)}{2}+|T|(n-|T|)}.\end{aligned}$$

Since there are $\binom{|S|}{\ell}$ subsets T of S of cardinal ℓ, we obtain

$$\Pr\left(\bigwedge_{i\in S} Z_i = 1\right) = 1 + \sum_{\ell=1}^{|S|} \binom{|S|}{\ell} (-1)^{\ell}(1-q_1)^{\ell} \cdot (1-q_2)^{\frac{\ell\cdot(\ell-1)}{2}+\ell\cdot(n-\ell)}.$$

□

D Proofs for Section 6

D.1 Proof of Proposition 8

Proof. By definition, an orbit is a closed path in the leakage diagram $\Gamma^p_{C_n,\Phi}$. In particular, every time the orbit leaves the circular layer of a gadget, it must use a non-horizontal edge, that is, an edge of type b or r, to reach the circular layer of another gadget. Moreover, since the path is closed, each such transition from one gadget to another must eventually be matched by another transition returning to the original gadget layer. Therefore, non-horizontal edges are necessarily used in pairs along the orbit. It follows that the total number of non-horizontal edges contained in O is even. □

D.2 Proof of Proposition 9

Proof. We prove the statement by induction on i.

Base case ($i = 1$). Assume that $k \geq n+1$. We show that O contains at least 2 non-horizontal edges. Suppose, for contradiction, that O contains at most one non-horizontal edge. By Proposition 8, the number of such edges must be even, and hence equal to 0. Therefore, O only uses horizontal edges (*i.e.*, edges internal to a gadget). In that case, the orbit is entirely contained within a single circular layer of a gadget, and its size is exactly n, which contradicts the assumption $k \geq n+1$.

Induction step. Assume that the statement holds for $i-1$, that is, any orbit of size $k \geq n+3(i-1)-2$ contains at least $2(i-1)$ non-horizontal edges. We now prove the result for i. Suppose, for contradiction, that O contains at most $2i-1$ non-horizontal edges. By Proposition 8, this number must be even, and hence at most $2(i-1)$. On the other hand, since $k \geq n+3i-2 > n+3(i-1)-2$, the induction hypothesis implies that O contains at least $2(i-1)$ such edges. Therefore, O contains exactly $2(i-1)$ non-horizontal edges.

We now upper-bound the size of such an orbit with $2(i-1)$ non-horizontal edges. More precisely, we show that

$$k \leq n+3(i-1).$$

These $2(i-1)$ non-horizontal edges can be arranged in two groups: one group of $i-1$ edges that are taken downwards and $i-1$ edges that are taken upwards. Besides, in each group, one can make pairs of one edge going downwards and one edge going upwards on the same refresh gadget. If there are many on the same refresh gadget, the number is still even and we create pairs by consecutive order. For each such pair, there are 3 cases, illustrated in Figure 9:

a If both edges are of type b or of type r, then the distance between the two edges is unchanged. Hence contributing by 2 to the size of the orbit.
b If the downward edge is of type r and the upward edge is of type b, then the distance between the two edges gets decreased by 1. Hence contributing by 1 to the size of the orbit.
c If the downward edge is of type b and the upward edge is of type r, then the distance between the two edges gets increased by 1. Hence contributing by 3 to the size of the orbit.

Hence, the largest orbit with $2(i-1)$ non-horizontal edges is the one where all pairs are of the last type, which contributes 3 to the size of the orbit. Therefore, we obtain

$$k \leq n+3(i-1) = n+3i-3 < n+3i-2,$$

which contradicts the assumption $k \geq n+3i-2$.

□

A useful reformulation of Proposition 9 is given in Corollary 2.

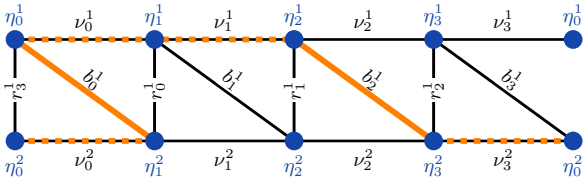
Corollary 2. *Let O be an orbit of size k in $\Gamma^p_{C_n,\Phi}$. Let $i \in [1, \lfloor k/2 \rfloor]$. If O contains exactly $2i$ non-horizontal edges* (i.e., *edges of type b or r), then*

$$k \leq n+3i.$$

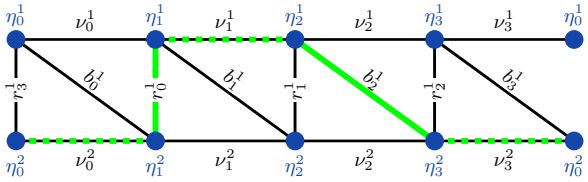
Proof. We prove the statement by contraposition. Suppose that $k > n+3i$. Then $k \geq n+3(i+1)-2$. By Proposition 9, this implies that O contains at least $2(i+1)$ non-horizontal edges, which contradicts the assumption that it contains exactly $2i$ such edges. □

D.3 Proof of Lemma 6

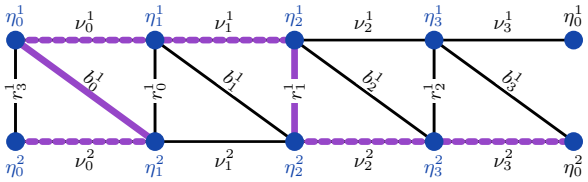
Proof. For each gadget G in C_n, let E_G denote the event that $\Gamma^p_{C_n,\Phi}$ contains an orbit of size $n+1$ associated to G, *i.e.*, an orbit of size $n+1$ that contains at least one edge of G and no edge of any preceding gadget.



(a) Example case a: the pair on the refresh layer contributes to the size of the orbit by a factor of 2.



(b) Example case b: the pair on the refresh layer contributes to the size of the orbit by a factor of 1.



(c) Example case c: the pair on the refresh layer contributes to the size of the orbit by a factor of 3.

Fig. 9: Three example cases for the contribution of the pair on the refresh layer to the size of the orbit. The dashed part corresponds to the part that is not studied here.

Then,

$$\begin{aligned}\Pr\Big[\Gamma^p_{C_n,\Phi} \text{ contains an orbit of size } n+1\Big] &= \Pr\Big[\bigcup_G E_G\Big] \\ &\le \sum_G \Pr[E_G],\end{aligned}$$

by a union bound. We now upper bound $\Pr[E_G]$ for a fixed gadget G. An orbit contributing to E_G may either (i) move *upward* towards preceding gadgets, or (ii) move *downward* towards subsequent gadgets. In both cases, the orbit can leave G only through edges of type b or r.

Downward case. Such orbits must contain one b-edge to move to the next gadget and one r-edge to return. Moreover, the return edge cannot be the one immediately preceding the chosen b-edge; otherwise, the orbit would not include any edge of G. This r-edge and b-edge contribute a factor at most $3p^2$ for the two non-horizontal edges. The remaining $n-1$ edges are distributed between G and the next gadget, depending on the choice of (b,r).

Since the type of the next gadget is not controlled, we upper bound its contribution by considering the worst case (a multiplication gadget), while using the corresponding probabilities for edges inside G. A factor 2 is also added when G is a copy since it doubles the choices of (b,r). This yields the first term of the bound.

Upward case. By construction, such orbits cannot contain a plateau: otherwise, they would visit an edge of a preceding gadget and would therefore not be associated to G. Hence, the orbit must immediately return to G through a pair (b,r) such that the index of b is the index of r plus 1. There are exactly n such pairs. Each such configuration requires one b-edge and one r-edge to belong to $\Gamma^p_{C_n,\Phi}$, which occurs with probability at most $p \cdot 3p = 3p^2$, and $n-1$ horizontal edges of G to belong to $\Gamma^p_{C_n,\Phi}$. Taking into account the possible choices of input gadgets (with a factor 2 for addition and multiplication gadgets), we obtain the second term of the bound.

Summing over all gadgets G concludes the proof. □

D.4 Proof of Lemma 7

Proof. For orbits of size $n+2$, we consider the eight following scenarios:

1. *Orbits going through two b-edges of the downward refresh gadget.* In this case, the orbit exits the current gadget and returns to it by using two distinct b-edges within the same downward refresh gadget. There are $\binom{n}{2}$ choices for the pair of b-edges. For each choice, two orbits are possible depending on whether the plateau lies on the current gadget or on the next one (illustrated by the green and red paths in Figure 10). Moreover, this pattern cannot move upward without creating a plateau; hence only the downward direction contributes. For copy gadgets (with two outputs), the contribution is doubled.
   Let b and b' such that $0 \le b < b' \le n-1$ denote the indices of the two b-edges, and set $\Delta = b' - b$. Depending on the plateau location, the orbit contains Δ edges in one gadget and $n-\Delta$ edges in the other.
   For the next gadget, we upper bound the contribution by considering the worst case, namely a multiplication gadget. Therefore, we obtain

$$p^2 \cdot \sum_{b=0}^{n-2} \sum_{b'=b+1}^{n-1} \Big(p_{\max}^{(\Delta)} \cdot \Big(\sum_{\alpha \in \{\text{add},\text{mult},\text{lin}\}} p_\alpha^{n-\Delta} \cdot |\mathcal{G}_\alpha| + 2 \cdot p_{\text{copy}}^{n-\Delta} \cdot |\mathcal{G}_{\text{copy}}| \Big) + p_{\max}^{(n-\Delta)} \cdot \Big(\sum_{\alpha \in \{\text{add},\text{mult},\text{lin}\}} p_\alpha^{\Delta} \cdot |\mathcal{G}_\alpha| + 2 \cdot p_{\text{copy}}^{\Delta} \cdot |\mathcal{G}_{\text{copy}}| \Big) \Big),$$

   where $\Delta = b' - b$.

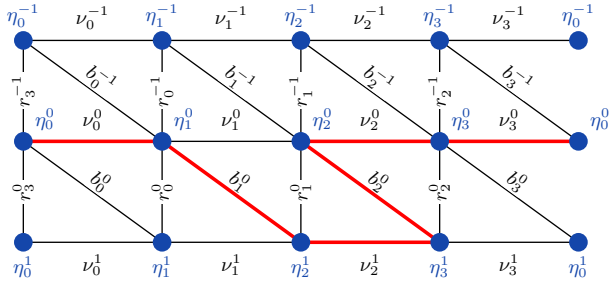


Fig. 10: Example of Case 1 for orbit of size $n+2$: a lower path using only b-edges.

2. *Orbits going through two r-edges of the downward refresh gadget.* This case is analogous to the previous one with two b-edges. The orbit leaves and returns to the current gadget using two distinct r-edges within the same downward refresh gadget. As before, there are $\binom{n}{2}$ choices, each yielding two possible orbits depending on the plateau location. The only difference lies in the leakage probability of r-edges, which is at most $3p$ instead of p. Therefore, we obtain the same bound as in the previous case, up to replacing p^2 by $(3p)^2$:

$$(3p)^2 \cdot \sum_{b=0}^{n-2} \sum_{b'=b+1}^{n-1} \Big(p_{\max}^{(\Delta)} \cdot \Big(\sum_{\alpha \in \{\text{add},\text{mult},\text{lin}\}} p_\alpha^{n-\Delta} \cdot |\mathcal{G}_\alpha| + 2 \cdot p_{\text{copy}}^{n-\Delta} \cdot |\mathcal{G}_{\text{copy}}| \Big) + p_{\max}^{(n-\Delta)} \cdot \Big(\sum_{\alpha \in \{\text{add},\text{mult},\text{lin}\}} p_\alpha^{\Delta} \cdot |\mathcal{G}_\alpha| + 2 \cdot p_{\text{copy}}^{\Delta} \cdot |\mathcal{G}_{\text{copy}}| \Big) \Big),$$

   where $\Delta = b' - b$.

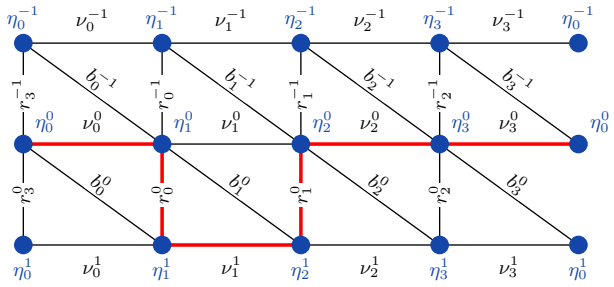


Fig. 11: Example of Case 2 for orbit of size $n+2$: a lower path using only r-edges.

3. *Orbits reaching two consecutive gadgets downward.* Such orbits are parameterized by tuples (b^0, b^1, r^0, r^1), where b^0 (resp. b^1) is the index of the b-edge used for the first (resp. second) downward move, and r^1 (resp. r^0) is the index of the r-edge used for the first (resp. second) return move. Valid orbits (*i.e.*, containing at least one edge of the current gadget) correspond to tuples (b^0, b^1, r^0, r^1) satisfying one of the following four relative orderings (excluding degenerate cases):

$$
\begin{aligned}
&0 \le b^0 < b^1 \le r^1 \le r^0 < n && \text{excluding } (b^0, r^0) = (0, n-1),\\
&0 \le r^0 < b^0 < b^1 \le r^1 < n && \text{excluding } r^0 = b^0 - 1,\\
&0 \le r^1 \le r^0 < b^0 < b^1 < n && \text{excluding } r^0 = b^0 - 1,\\
&0 \le b^1 \le r^1 \le r^0 < b^0 < n && \text{excluding } r^0 = b^0 - 1.
\end{aligned}
$$

Figure 12 illustrates these four scenarios.

Altogether, this yields $n \cdot \binom{n}{3}$ admissible tuples. For copy gadgets, we must additionally account for the two possible outputs. Moreover, when considering the gadgets reached after exiting the current one, we upper bound the probability by assuming the worst case, namely that all subsequent gadgets are copy gadgets, which introduces an additional factor 2 (in practice, they can also be 2-to-1 gadgets, making it possible to reach their second input).

Taking into account the number of edges in the current gadget (denoted by Δ) and in the subsequent gadgets (which we pessimistically assume to be multiplication gadgets, as this yields the largest contribution), we obtain:

$$
\begin{aligned}
2 \cdot (3p^2)^2 \cdot n \cdot \sum_{\Delta=1}^{n-2} \sum_{b^1=1}^{n-\Delta-1} (n - \Delta - b^1) \cdot p_{\max}^{(n-2-\Delta)} \\
\cdot \sum_{\alpha \in \{\text{add},\text{mult},\text{lin}\}} \left(p_\alpha^\Delta \cdot |\mathcal{G}_\alpha| \; + \; 2 \cdot p_{\text{copy}}^\Delta \cdot |\mathcal{G}_{\text{copy}}| \right).
\end{aligned}
$$

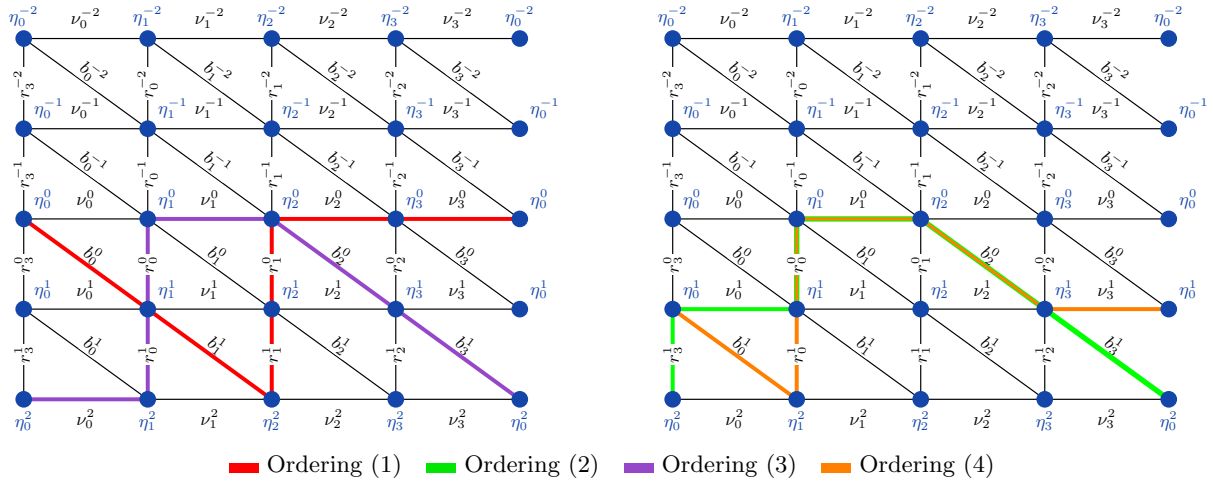


Fig. 12: Example of Case 3 for orbits of size $n+2$: downward round trips with possible plateaus.

4. *Orbits reaching two consecutive gadgets upward.* Such orbits require two r-edges of indices (r^{-1}, r^{-2}) to move upward and two b-edges of indices (b^{-2}, b^{-1}) to return downward. Since plateaux are excluded by construction in this case, the two upward steps must use the same r-index, *i.e.*, $r^{-1} = r^{-2}$, and the return must satisfy $b^{-1} = b^{-2} + 1 = r^{-1} + 2$ (modulo n). Hence there are exactly n admissible (r, b) choices. All remaining $n-2$ edges of the orbit lie within the current gadget. We must consider a factor of 2 for 2-input gadgets and an additional factor of 2 in the pessimistic case where the previous gadget also has two inputs. An example of such orbits is shown in Figure 13.
   Therefore, we obtain

$$n \cdot (3p^2)^2 \cdot 2 \cdot \left(2 \cdot p_{\text{mult}}^{n-2} \cdot |\mathcal{G}_{\text{mult}}| + 2 \cdot p_{\text{add}}^{n-2} \cdot |\mathcal{G}_{\text{add}}| + p_{\text{copy}}^{n-2} \cdot |\mathcal{G}_{\text{copy}}| + p_{\text{lin}}^{n-2} \cdot |\mathcal{G}_{\text{lin}}|\right).$$

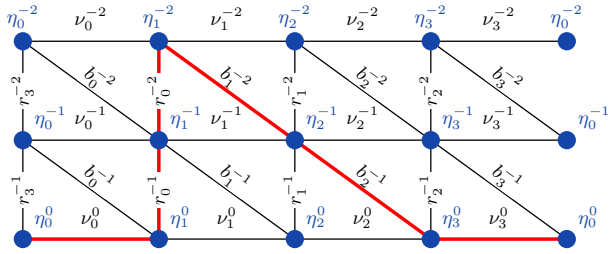


Fig. 13: Example of Case 4 for orbits of size n+2: upward round trips without plateau.

5. *Orbits reaching one gadget downward (or two if the current gadget is a copy gadget).* These orbits consist of two round trips going downward, possibly with plateaus (*i.e.*, one refresh gadget downward, or two in the case of copy gadgets). They fall into two distinct classes, characterized by the relative ordering of the indices.
   In the first class, the indices satisfy

$$0 \le r^0 < b^0 \le r^1 < b^1 < n,$$

which corresponds to $\binom{n}{3} + \binom{n}{4}$ possibilities and is illustrated in red in Figure 14. Among these, $\binom{n-1}{1}$ configurations correspond to orbits that do not contain any edge of the current gadget G and must therefore be excluded.
The contribution of this first class is thus given by

$$\sum_{r^0=0}^{n-1} \sum_{b^0=r^0+1}^{n-1} \sum_{r^1=b^0}^{n-1} \sum_{\substack{b^1=r^1+1 \\ (b^0,b^1)\neq(r^0+1,r^1+1)}}^{n-1} (3p^2)^2 \cdot p_{\max}^{(n-2-\Delta)} \cdot \sum_{\alpha\in\{\text{add,mult,lin}\}} \Big(p_\alpha^\Delta \cdot |\mathcal{G}_\alpha| + 4 \cdot p_{\text{copy}}^\Delta \cdot |\mathcal{G}_{\text{copy}}|\Big),$$

where

$$\Delta = b^0 - r^0 + b^1 - r^1 - 2.$$

In the second class, the indices satisfy

$$0 \le b^0 \le r^0 < b^1 \le r^1 < n,$$

which yields $\binom{n+2}{4}$ possibilities and is illustrated in green in Figure 14. In this case, $\binom{n-1}{2}$ configurations correspond to orbits without any edge in G and are excluded.
The contribution of the second class is therefore

$$\sum_{b^0=0}^{n-1} \sum_{r^0=b^0}^{n-1} \sum_{b^1=r^0+1}^{n-1} \sum_{\substack{r^1=b^1 \\ (b^0,r^0,b^1,r^1)\neq(0,r^0,r^0+1,n-1)}}^{n-1} (3p^2)^2 \cdot p_{\max}^{(\Delta)} \cdot \sum_{\alpha\in\{\text{add,mult,lin}\}} \Big(p_\alpha^{n-2-\Delta} \cdot |\mathcal{G}_\alpha| + 4 \cdot p_{\text{copy}}^{n-2-\Delta} \cdot |\mathcal{G}_{\text{copy}}|\Big),$$

with

$$\Delta = r^0 - b^0 + r^1 - b^1.$$

In both cases, we must additionally account for the two possible outputs of each copy gadget, yielding a factor 4 to select the outputs for the two round trips.

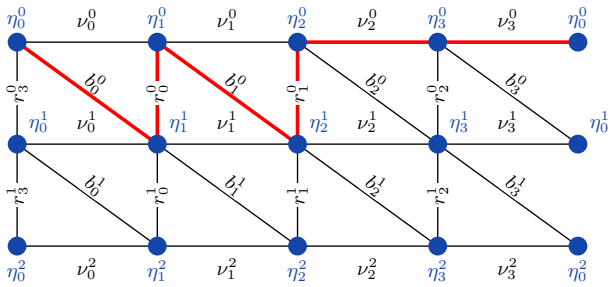


Fig. 14: Example of case 5 for orbit of size n+2: two downward round trips with possible plateaus

6. *Orbits reaching one gadget upward (or two if two inputs).* This case corresponds to two consecutive upward round trips, either through a single upward refresh gadget, or through two upward refresh gadgets when the current gadget has several inputs. Without a plateau, each round trip must use a compatible pair (b, r) with matching indices. For the single-refresh configuration, there are $\binom{n}{2}$ ways to

choose the two (distinct) index pairs (one such orbit is illustrated in Figure 15). In addition, for copy gadgets, a second configuration is possible where one such pair is chosen independently for each input, yielding n^2 possibilities. Therefore, we obtain

$$\binom{n}{2} \cdot (3p^2)^2 \cdot \Big(\sum_{\alpha \in \{\text{add,mult,lin}\}} p_\alpha^{n-2} \cdot |\mathcal{G}_\alpha| + 2 \cdot p_{\text{copy}}^{n-2} \cdot |\mathcal{G}_{\text{copy}}| \Big) + n^2 \cdot (3p^2)^2 \cdot p_{\text{copy}}^{n-2} \cdot |\mathcal{G}_{\text{copy}}|.$$

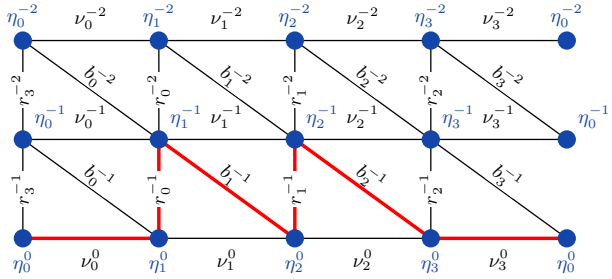


Fig. 15: Example of case 6 for orbit of size n+2: two upward round trips without plateaus.

7. *Orbits reaching one gadget downward and one gadget upward.* In this scenario, the orbit reaches one subsequent gadget and one preceding gadget. The upward step cannot contain a plateau, hence it is uniquely determined by a compatible (r^{-1}, b^{-1}) pair, giving exactly n possible choices. Illustrations of such orbits are provided in Figure 16.

   Conditioning on this upward pair, the downward step must use a pair (b^0, r^0) with $b^0 \le r^0$ among the remaining $n-1$ indices. We must exclude the unique degenerate choice where (b^0, r^0) meets the already chosen upward edges (b^{-1}, r^{-1}), since in that case the orbit would not traverse any edge of the current gadget. Therefore, for a fixed upward pair, the number of admissible downward pairs is

$$\#\{(b^0, r^0) : 0 \le b^0 \le r^0 \le n-2\} - 1 = \left(\binom{n-1}{2} + (n-1)\right) - 1 = \binom{n}{2} - 1.$$

   Multiplying the n choices for the upward pair and accounting for the leakage probabilities yields:

$$n \cdot (n-1) \cdot (3p^2)^2 \cdot \left(\sum_{\alpha \in \{\text{add,mult,copy}\}} 2 \cdot p_\alpha^{n-2} \cdot |\mathcal{G}_\alpha| \; + \; p_{\text{lin}}^{n-2} \cdot |\mathcal{G}_{\text{lin}}| \right) + n \cdot (3p^2)^2 \cdot \sum_{\substack{0 \le b^0 < r^0 \le n-2 \\ (b^0, r^0) \ne (0, n-2)}} p_{\max}^{(r^0 - b^0)} \cdot \left(\sum_{\alpha \in \{\text{add,mult,copy}\}} 2 \cdot p_\alpha^{n-2-(r^0-b^0)} \cdot |\mathcal{G}_\alpha| \; + \; p_{\text{lin}}^{n-2-(r^0-b^0)} \cdot |\mathcal{G}_{\text{lin}}| \right).$$

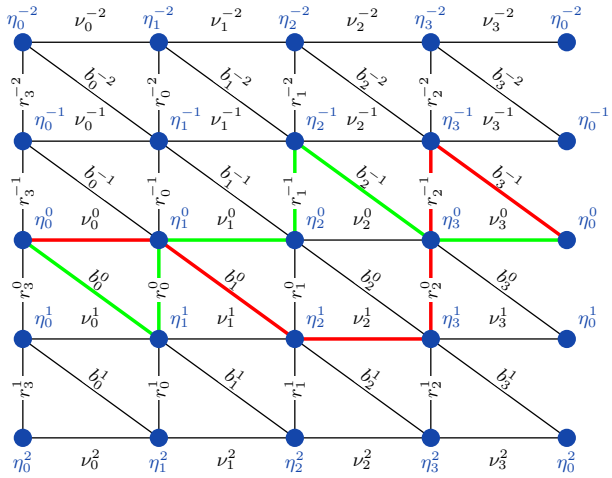


Fig. 16: Example of Case 7 for orbits of size n+2: one downward round trip with possible plateau and one upward round trip without plateau.

8. *Orbits reaching one copy gadget upward and its second output.* This case is similar to the one where the orbit reaches two consecutive gadgets upward, except that we now allow one plateau on the second output of a preceding copy gadget. The first upward round trip is determined by a pair of edges (r^{-1}, b^{-1}) that must make such a plateau possible. This requires that the indices satisfy one of the following two conditions:

$$\begin{aligned} &0 \le r^{-1} < b^{-1} < n, \qquad && b^{-1} - r^{-1} \ge 3, \\ &0 \le b^{-1} < r^{-1} < n, \qquad && n - r^{-1} + b^{-1} \ge 3. \end{aligned}$$

Once this first pair is fixed, the second pair $(r^{-1'}, b^{-1'})$ leading to the second output of the copy gadget is uniquely determined to avoid a plateau on the copy gadget.

We now count the admissible choices for (r^{-1}, b^{-1}). For a fixed r^{-1}, there are exactly $n-3$ admissible values of b^{-1}, so the total number of such pairs is $n(n-3)$. Each resulting orbit contains two r-edges and two b-edges, contributing a factor $(3p^2)^2$, together with $n-2$ horizontal edges split into the current gadget and the second output of the copy gadget. Since the reached gadget is a copy gadget, we obtain the contribution

$$(3p^2)^2\, n \sum_{d=3}^{n-1} p_{\max}^{(d-2)} \left(2 \sum_{\alpha \in \{\text{add,mult}\}} p_\alpha^{n-d} |\mathcal{G}_\alpha| + \sum_{\alpha \in \{\text{lin,copy}\}} p_\alpha^{n-d} |\mathcal{G}_\alpha| \right).$$

where the factor 2 accounts for the two possible inputs of the current gadget.

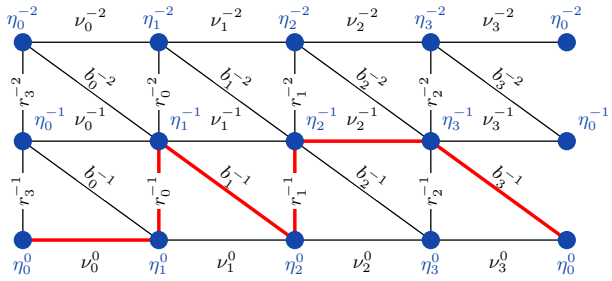


Fig. 17: Example of case 8 for orbit of size n+2: one upward round trip with plateau on the second output of a copy gadget and one upward round trip without plateau.

□

D.5 Proof of Lemma 8

Proof. A naive upper bound on the probability that there exists an orbit of size at least $n+3$ can be obtained by counting all possible edge combinations, yielding

$$|\mathcal{G}| \cdot (12 \cdot p + 2 \cdot p_{\max})^{n+3}. \tag{1}$$

However, this bound is overly pessimistic, as it treats all edge combinations as feasible. In particular, not all configurations of edges can form a valid orbit, and among those that do, edges of certain types typically occur with lower probability. By exploiting these structural constraints, we can discard a large number of impossible or unlikely configurations and derive a significantly tighter bound. Let

$$E_{k,i} = \left\{ \Gamma^{p}_{C_n,\Phi} \text{ contains an orbit of size } k \text{ with exactly } 2i \text{ non-horizontal edges} \right\}.$$

Then we start from the following decomposition:

$$\begin{aligned}
&\Pr\left[\Gamma^{p}_{C_n,\Phi} \text{ contains an orbit of size at least } n+3\right] \\
&\leq \sum_{k=n+3}^{s} \Pr\left[\Gamma^{p}_{C_n,\Phi} \text{ contains an orbit of size } k\right] \\
&\leq \sum_{k=n+3}^{s} \sum_{i=0}^{\lfloor k/2 \rfloor} \Pr\left[E_{k,i}\right] \qquad \text{by Proposition 8} \\
&\leq \sum_{k=n+3}^{s} \sum_{i=\lceil (k-n)/3 \rceil}^{\lfloor k/2 \rfloor} \Pr\left[E_{k,i}\right] \qquad \text{by Corollary 2} \\
&= \Pr\left[E_{n+3,1}\right] + \sum_{k=n+3}^{s} \sum_{i=\max(\lceil (k-n)/3 \rceil, 2)}^{\lfloor k/2 \rfloor} \Pr\left[E_{k,i}\right] \\
&\leq \underbrace{\Pr\left[E_{n+3,1}\right]}_{①} + \sum_{k=n+3}^{\beta} \sum_{i=\max(\lceil (k-n)/3 \rceil, 2)}^{\lfloor k/2 \rfloor} \underbrace{\Pr\left[E_{k,i}\right]}_{②} + |\mathcal{G}|(12p + 2p_{\max})^{\beta+1},
\end{aligned}$$

where the last inequality uses the naive bound from Equation 1 for all orbit sizes larger than β. Let us first bound the term ①, corresponding to orbits of size $n+3$ with exactly two non-horizontal edges (see Figure 18).

As in the cases $n+1$ and $n+2$, we count such orbits gadget by gadget, restricting to downward orbits, since upward ones would necessarily introduce a plateau. For $\Delta \in [1, n-1]$, the orbit consists of $\Delta+1$ consecutive horizontal edges in the current gadget and $n-\Delta$ in the next one.

To obtain an orbit of size $n+3$, the orbit must exit the current gadget through an r-edge (with probability at most $3p$) and return through a b-edge (with probability at most p). The choice of the r-edge determines the starting position and contributes a factor n (or $2n$ for copy gadgets as there are two possible directions). Finally, the $n-\Delta$ horizontal edges lie in the next gadget, whose contribution is upper bounded by considering the worst case, namely a multiplication gadget.

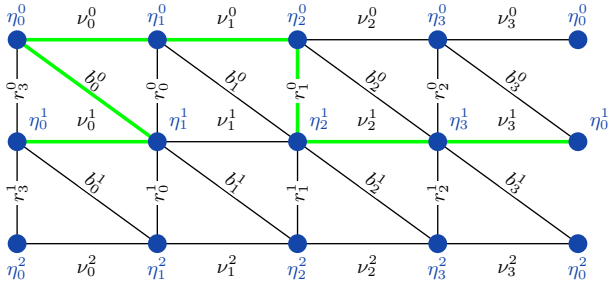


Fig. 18: Example of the second-term orbits of size at least n+3 with exactly two (r, b)-edges.

$$\textcircled{1}: \quad \Pr\left[E_{n+3,1}\right] \;\leq\; n \cdot (3p^2) \cdot \sum_{\Delta=1}^{n-2} p_{\max}^{(n-\Delta)} \left(\sum_{\alpha \in \{\text{add,mult,lin}\}} p_\alpha^{\Delta+1} \cdot |\mathcal{G}_\alpha| + 2 \cdot p_{\text{copy}}^{\Delta+1} \cdot |\mathcal{G}_{\text{copy}}| \right).$$

Let us now bound the term ②, corresponding to orbits of size $k \geq n+3$ with exactly $2i$ non-horizontal edges. A first (coarse) bound can be obtained by counting all possible placements of these edges along the orbit. Among the k edges of the orbit, exactly $2i$ are non-horizontal (of type (r,b)), while the remaining $k-2i$ are horizontal edges. In the worst case, we upper bound these horizontal edges by multiplication edges. There are $\binom{k}{2i}$ ways to choose the positions of the $2i$ non-horizontal edges. Each such edge appears with probability at most $12p$ (accounting for the worst case with up to three inputs/outputs per gadget), while each horizontal edge appears with probability at most $2p_{\max}$. Therefore,

$$\textcircled{2}: \quad \Pr\left[E_{k,i}\right] \;\leq\; |\mathcal{G}| \binom{k}{2i} (12p)^{2i} (2p_{\max})^{k-2i}.$$

However, we aim at a refined bound. We estimate the bound by counting the orbits gadget by gadget assuming that the orbit touches at least one point of the current gadget. We refine the bound depending on whether the type of the current gadget.

1. If the current gadget is a multiplication, the bound of the previous equation holds.
2. If the current gadget is an addition or copy gadget, we can refine the bound a bit more. Indeed, before leaving the gadget, the horizontal edges must have the probability p_{add} or p_{copy} respectively. Similarly, when coming back to the gadget, the horizontal edges must have the probability p_{add} or p_{copy} respectively. Therefore, let L be the number of edges between the first exit and the last return. There are $k-L$ horizontal edges in the gadget that have the probability p_{add} or p_{copy}. The other $L-2i$ edges have the probability $p_{\max}$ (worst case). Once L is fixed, there are $(k-L+1)$ ways to choose the position of the L edges between the first exit and the last return. Among these L edges, there are $\binom{L-2}{2i-2}$ ways to choose the $2i$ (r,b)-edges (as the first exit and last return are (r,b)-edges).

3. If the current gadget is an linear gadget, we can refine the bound even more. Indeed, among the $2i$ (r,b)-edges of the orbit, at least two of them are the exit and the return (r,b) edges of the current gadget. Since, the gadget is linear, for the exit and return, instead of $12p$ there is a probability of $8p$.

$$\begin{aligned}
\text{② (refined)}: \quad \Pr\big[E_{k,i}\big] \;\le\; & |\mathcal{G}_{\text{mult}}| \binom{k}{2i} (12p)^{2i} (2p_{\text{mult}})^{k-2i} \\
& + \sum_{L=\max(2i,k-n)}^{k} (k-L+1) \binom{L-2}{2i-2} (2p_{\max})^{L-2i} \\
& \quad \cdot \Big[(12p)^{2i} \big(|\mathcal{G}_{\text{add}}| (2p_{\text{add}})^{k-L} + |\mathcal{G}_{\text{copy}}| (2p_{\text{copy}})^{k-L} \big) \\
& \qquad + (8p)^2 (12p)^{2i-2} |\mathcal{G}_{\text{lin}}| (2p_{\text{lin}})^{k-L} \Big].
\end{aligned}$$

This leads to the final bound. □