Compositional Security for Task-PIOAs

Ran Canetti[1,2], Ling Cheung[2,3], Dilsun Kaynar[2,4],
Nancy Lynch[2], and Olivier Pereira[5]

[1] IBM TJ Watson Research Center
[2] Massachusetts Institute of Technology
[3] Radboud University of Nijmegen
[4] Carnegie Mellon University
[5] Université catholique de Louvain, olivier.pereira@uclouvain.be

Abstract. Task-PIOA is a modeling framework for distributed systems with both probabilistic and nondeterministic behaviors. It is suitable for cryptographic applications because its task-based scheduling mechanism is less powerful than the traditional perfect-information scheduler. Moreover, one can speak of two types of complexity restrictions: time bounds on description of task-PIOAs and time bounds on length of schedules. This distinction, along with the flexibility of nondeterministic specifications, are interesting departures from existing formal frameworks for computational security.

The current paper presents a new approximate implementation relation for task-PIOAs. This relation is transitive and is preserved under hiding of external actions. Also, it is shown to be preserved under concurrent composition, with any polynomial number of substitutions. Building upon this foundation, we present the notion of structures, which classifies communications into two categories: those with a distinguisher environment and those with an adversary. We then formulate secure emulation in the spirit of traditional simulation-based security, and a composition theorem follows as a corollary of the composition theorem for the new approximate implementation relation.

1 Introduction

Cryptographic protocols are distributed algorithms that must achieve security properties such as authentication and secret communication, while operating in environments that include adversarial components. Security and correctness of such protocols can be vital to the survival of commercial and military enterprises. However, many cryptographic protocols exhibit complex, subtle behavior, so verifying their security is not easy. Informal verification is not reliable enough; what is needed is a set of rigorous, formal verification methods that can assert protocol security and correctness, while being reasonably easy for protocol designers to use.

One of the main sources for intricacies in security analysis of these protocols is the fact that in most interesting cases security can hold only in a "computational sense", namely only against computationally bounded adversaries, only probabilistically, and only under computational hardness assumptions. Current security analyses of protocols deal with this issue in one of two ways. One way is to first analyze the protocol in an idealized model where cryptographic algorithms are represented via symbolic operations and security assertions can be absolute rather than "computational" (e.g., [1–9]); then, additional steps are taken outside the formal model to provide security guarantees when the symbolic operations are replaced by real algorithms (e.g., [10–12]).

An alternative approach is to extend the formal model so as to directly capture "computational security" within the model itself. This requires representing within the model resource bounded, probabilistic computations as well as probabilistic relations between systems and system components. Such models include Probabilistic Polynomial-Time Process Calculus (PPC) [13–15], Reactive Simulatability (RSIM) [16–18], Universally Composable (UC) Security [19], Task-PIOA [20, 21] and Inexhaustible Interactive Turing Machine (IITM) [22]. Each of these frameworks can be decomposed into two "layers": (i) a foundational layer, which consists of a general model of concurrent computation with time bounds, not specific to security protocols, and (ii) a security layer that typically follows the general outline of simulation-based security [23–29]. Unlike the security layer, the foundation layer varies widely across different frameworks. We summarize a few main differences below.

Description of concurrent processes. PPC is process theoretic, RSIM and Task-PIOA are based on abstract state machines, and UC and IITM are based on interactive Turing machines. In RSIM, UC and IITM, machines are purely probabilistic, meaning that their behaviors are completely determined up to inputs and coin tosses. In contrast, PPC and Task-PIOA allow nondeterministic process specifications. More detailed comparisons of Task-PIOA against PPC and RSIM can be found in the latest version of [30].

Sequential vs. non-sequential scheduling. The two ITM-based frameworks, UC and ITTM, use sequential scheduling. This means machines are activated in succession, where the current active machine triggers the next one by sending

a message. RSIM machines use a similar mechanism, but with special "buffer" machines to capture message delays and "clock ports" to control the scheduling of message delivery. Hence, non-sequential scheduling may be implemented in RSIM; however, in actual protocol analysis, sequential scheduling is typically used (e.g., [31]). With the exception of its sequential variant [32], PPC implements non-sequential scheduling with scheduler functions (or Markov chains) that select the next action from a set of enabled actions. Task-PIOA is also non-sequential, using arbitrary oblivious task sequences to determine the next transition. We refer to [33] for examples showing that the choice between sequential and non-sequential scheduling leads to different notions of simulation-based security.

Complexity bounds. In PPC, processes are finite expressions built up from a grammar that contains bounded replication operators $!_{q(k)}$, where k is the security parameter and q is a polynomial. Given any process $\mathcal{P}$, $!_{q(k)}(\mathcal{P})$ is evaluated as $q(k)$ copies of $\mathcal{P}$ in parallel. It is proven in [15] that every variable-closed process expression can be evaluated in time polynomial in the security parameter. In RSIM, abstract machines are realized by Turing machines that are either polynomial time in the security parameter or in the overall length of inputs, although major results such as composition theorems are proven only for the former notion of polynomial time. In UC and IITM, ITMs may have runtime polynomial in the overall length of inputs, provided certain restrictions are observed. These restrictions make sure that the runtime of an entire system is polynomial in the security parameter.

Task-PIOA occupies an interesting middle ground in the treatment of time bounds. Each task-PIOA[1] must have description bounded by a polynomial in the security parameter. This applies to the representations of states, actions, transitions, etc. In addition, the transition relation must be computable by a probabilistic Turing machine with runtime polynomial in the security parameter. However, there is no *a priori* bound imposed on the number of transitions that a task-PIOA may perform. Hence, a task-PIOA specification has potentially unbounded behavior. A final restriction on runtime is imposed only when we compare the behaviors of different task-PIOAs using implementation relations.

We believe it is meaningful to consider these two types of time bounds separately, since they express limitations of different nature. For example, in modeling *long-lived* security protocols [34], limitations on what a machine can do in one step (or in a bounded amount of time) are quite different from limitations on the total lifetime of the machine.

Also, as illustrated in [33], this separation of time bounds allows us to define unbounded forwarders without any additional mechanism, such as the input guards of [32, 22]. (As shown in [32], the existence of forwarders has a great impact on the relationships between different notions of simulated-security.) Nor do we need to face the usual hassles associated with ITMs that are polynomial time in the overall length of inputs. That is, we do not need to impose special

[1] Technically, we should refer to task-PIOA families. We omit "families" for simplicity.

restrictions, such as those in UC and IITM, to make sure that computation resources are not "created" excessively as machines send inputs to each other.

1.1 Composability of Secure Emulation

A notable advantage of simulation-based security is its potential *security preserving composability* properties. Indeed, one of the main motivations behind the PPC, RSIM and UC frameworks was to obtain a very general composition operation that is provably security-preserving.

In a previous case study [20], we followed closely the setup of simulation-based security, and, in a more recent paper [33], we gave a generic formulation of *secure emulation* in the Task-PIOA framework. The main goal of this paper is to prove a polynomial composition theorem for our notion of secure emulation. While such theorems have been obtained in many of the aforementioned frameworks [19, 14, 35, 22], our version is interesting in its own right.

First of all, as pointed out in [33], the choice between sequential and non-sequential scheduling schemes gives rise to incomparable notions of security. In other words, even if we use the same high-level formulation of security, there exist protocols that are secure under sequential scheduling but not under non-sequential scheduling, and vice versa. Since Task-PIOA uses non-sequential scheduling, our composition theorem is *not* a simple transposition of composition theorems in sequential frameworks.

Secondly, our secure emulation is defined in terms of a new approximate implementation relation ($\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$) for task-PIOAs. As a result, our composition proof consists of two layers: we first prove a polynomial composition theorem for $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$, and the composition theorem for secure emulation follows as a corollary. Interestingly, the typical hybrid argument[2] is used in proving compositionality of $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$, which is completely independent of our formulation of secure emulation.

Finally, since the task-PIOA framework allows nondeterministic specifications with potentially unbounded behavior, we must handle two additional layers of quantifications while constructing a hybrid argument. (One of these involves schedule length bounds, while the other involves the resolution of nondeterminism.) In fact, compared to the definition of approximate implementation given in [20, 21], the definition of $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$ has a number of features inspired by the general structure of hybrid arguments. We refer to Section 3 for further discussions.

We now outline our formulation of secure emulation. Following [35], we introduce the notion of structures, which classifies communications into two categories: those with a distinguisher environment and those with an adversary. The former can be likened to I/O tapes in ITM-based frameworks and service ports in RSIM, while the latter can be likened to communication tapes and forbidden ports. We then define secure emulation to say roughly the following: a protocol ρ securely emulates a protocol ϕ if, for every adversary Adv for ρ, there is an

[2] Hybrid arguments are used widely in cryptography to handle polynomial growth in the number of composed protocols. We refer to [36] for an original description.

adversary Sim for ϕ such that the composition $\rho \| Adv$ implements the composition $\phi \| Sim$ in the sense of $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$. Note that every task-PIOA mentioned here has polynomially bounded description, but potentially unbounded runtime. The quantification over runtime bounds (i.e., schedule length bounds) are encapsulated in the definition of $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$. Moreover, the communications between ρ and Adv and between ϕ and Sim are hidden from the environment.

We prove that secure emulation, thus defined, is indeed compositional under a polynomial number of substitutions. This follows essentially as a corollary of the composition theorem for $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$. We also prove that secure emulation is transitive and preserved under hiding. These three properties, as well as invariant assertion and simulation relation techniques developed in [20, 37, 21, 30], are very beneficial for the scalability of computational analysis. For example, the composition theorem delineates situations in which multiple security protocols are run in parallel and we would like to prove that the security guarantees of individual component protocols are preserved in some appropriate sense. Also, we may specify protocols at different levels of abstraction, and use simulation relations to relate formally probability distributions on states (or executions) at adjacent levels. Such techniques make up a practical discipline of verification, since real-life security protocols operate not in isolation, but in the context of larger systems.

Overview Section 2 summarizes the task-PIOA framework presented in [37, 21]. In Section 3, we review the approximate implementation definition proposed in [20, 21], and introduce a new, stronger version of this definition, for which we present a polynomial composition theorem. We then provide a generic template for the use of task-PIOAs in cryptographic protocol specification, by defining the notions of structure and adversary for structures in Section 4. Equipped with these definitions, we define secure emulation in Section 5, and show it is preserved under polynomial composition.

2 Task-PIOAs

In this section, we review basic definitions in the Task-PIOA framework [37, 30]. We begin with the PIOA framework, which is a simple combination of I/O Automata [38] and Probabilistic Automata [39]. This is then augmented with a partial-information scheduling mechanism based on tasks. Finally, we bring in the notion of time bounds and its extension to task-PIOA families.

2.1 PIOAs

A *probabilistic I/O automaton (PIOA)* $\mathcal{A}$ is a tuple $\langle Q, \bar{q}, I, O, H, \Delta \rangle$, where: (i) Q is a countable set of *states*, with *start state* $\bar{q} \in Q$; (ii) I, O and H are countable and pairwise disjoint sets of actions, referred to as *input, output and internal actions*, respectively; (iii) $\Delta \subseteq Q \times (I \cup O \cup H) \times \mathsf{Disc}(Q)$ is a *transition relation*, where $\mathsf{Disc}(Q)$ is the set of discrete probability measures on Q. An action

a is *enabled* in a state q if $\langle q, a, \mu\rangle \in \Delta$ for some μ. The set $Act := I \cup O \cup H$ is called the *action alphabet* of $\mathcal{A}$. If $I = \emptyset$, then $\mathcal{A}$ is said to be *closed*. The set of *external* actions of $\mathcal{A}$ is $I \cup O$ and the set of *locally controlled* actions is $O \cup H$. Any sequence β of external actions is called a *trace*.

We require that $\mathcal{A}$ satisfies the following conditions.

- **Input Enabling:** For every $q \in Q$ and $a \in I$, a is enabled in q.
- **Transition Determinism:** For every $q \in Q$ and $a \in A$, there is at most one $\mu \in \mathsf{Disc}(Q)$ such that $\langle q, a, \mu\rangle \in \Delta$.

Parallel composition for PIOAs is based on synchronization of shared actions. PIOAs $\mathcal{A}_1$ and $\mathcal{A}_2$ are said to be *compatible* if $Act_i \cap H_j = O_i \cap O_j = \emptyset$ whenever $i \neq j$. In that case, we define their *composition* $\mathcal{A}_1 \| \mathcal{A}_2$ to be

$$\langle Q_1 \times Q_2, \langle \bar{q}_1, \bar{q}_2\rangle, (I_1 \cup I_2) \setminus (O_1 \cup O_2), O_1 \cup O_2, H_1 \cup H_2, \Delta\rangle,$$

where Δ is the set of triples $\langle\langle q_1, q_2\rangle, a, \mu_1 \times \mu_2\rangle$ such that (i) a is enabled in some q_i and (ii) for every i, if $a \in A_i$ then $\langle q_i, a, \mu_i\rangle \in \Delta_i$, otherwise μ_i assigns probability 1 to q_i (i.e., μ_i is the *Dirac* measure on q_i, denoted $\delta(q_i)$). Note that this definition of composition can be generalized to any finite number of components.

A *hiding* operator is also available: given $\mathcal{A} = \langle Q, \bar{q}, I, O, H, \Delta\rangle$ and $S \subseteq O$, $\mathsf{hide}(\mathcal{A}, S)$ is the tuple $\langle Q, \bar{q}, I, O', H', \Delta\rangle$, where $O' = O \setminus S$ and $H' = H \cup S$. Due to the compatibility requirement for parallel composition, the hiding operation prevents any other PIOA from synchronizing with $\mathcal{A}$ via actions in S.

2.2 Task-PIOAs

To resolve nondeterminism, we make use of the notion of tasks [38, 37]. Formally, a *task-PIOA* is a pair $(\mathcal{A}, \mathcal{R})$ such that (i) $\mathcal{A}$ is a PIOA and (ii) $\mathcal{R}$ is a partition of the locally-controlled actions of $\mathcal{A}$. With slight abuse of notation, we use $\mathcal{A}$ to refer to both the task-PIOA and the underlying PIOA. The equivalence classes in $\mathcal{R}$ are referred to as *tasks*. Unless otherwise stated, we will use terminologies inherited from the PIOA setting. The following axiom is imposed on task-PIOAs.

- **Action Determinism:** For every state $q \in Q$ and every task $T \in \mathcal{R}$, there is at most one action $a \in T$ that is enabled in q.

In case some $a \in T$ is enabled in q, we say that T is *enabled* in q.

Given compatible task-PIOAs $\mathcal{A}_1$ and $\mathcal{A}_2$, we define their *composition* to be $\langle \mathcal{A}_1 \| \mathcal{A}_2, \mathcal{R}_1 \cup \mathcal{R}_2\rangle$. Note that $\mathcal{R}_1 \cup \mathcal{R}_2$ is an equivalence relation because compatibility requires disjoint sets of locally controlled actions. It is also easy to check that action determinism is preserved under composition. The hiding operator for PIOAs extends in the obvious way: given a set S of output actions, $hide(\langle \mathcal{A}, \mathcal{R}\rangle, S)$ is simply $\langle hide(\mathcal{A}, S), \mathcal{R}\rangle$.

A *task schedule* for a closed task-PIOA $\langle \mathcal{A}, \mathcal{R}\rangle$ is a finite or infinite sequence $\rho = T_1.T_2.T_3 \ldots$ of tasks in $\mathcal{R}$. This induces a well-defined run of $\mathcal{A}$ as follows:

(i) from the start state $\bar{q}$, we consider the first task T_1;
(ii) due to action- and transition-determinism, T_1 specifies at most one transition from $\bar{q}$;

(iii) if such transition exists, it is taken, otherwise nothing happens;
(iv) repeat with remaining T_i's.

Such a run gives rise to a unique *trace distribution* of $\mathcal{A}$ (which is a probability distribution on the set of traces). The set of trace distributions induced by all possibly task schedules for $\mathcal{A}$ is denoted $\mathsf{TrDists}(\mathcal{A})$, while the trace distribution induced by the task schedule ρ for $\mathcal{A}$ is denoted $\mathsf{tdist}(\mathcal{A}, \rho)$. We refer to [30] for more details on trace distributions.

2.3 Time Bounds and Task-PIOA Families

In order to carry out computational analysis, we consider task-PIOAs whose operations can be represented by a collection of Turing machines with bounded run time. This is the Time-Bounded Task-PIOA model introduced in [21, 20].

We assume a standard bit-string representation for various constituents of a task-PIOA, including states, actions, transitions and tasks. Let $p \in \mathbb{N}$ be given. A task-PIOA $\mathcal{A}$ is said to have *p-bounded description* just in case:

(i) the representation of every constituent of $\mathcal{A}$ has length at most p;
(ii) there is a Turing machine that decides whether a given bit string is the representation of some constituent of $\mathcal{A}$;
(iii) there is a Turing machine that, given a state and a task of $\mathcal{A}$, determines the next action;
(iv) there is a probabilistic Turing machine that, given a state and an action of $\mathcal{A}$, determines the next state of $\mathcal{A}$;
(v) all these Turing machines can be described using a bit string of length at most p, according to some standard encoding of Turing machines;
(vi) all these Turing machines return after at most p steps on every input.

Thus, p limits the size of action names, the amount of available memory and the number of Turing machine steps taken at each transition of $\mathcal{A}$. It, however, does *not* limit the number of transitions that are taken in a particular run.

Suppose we have a compatible set $\{\mathcal{A}_i | 1 \leq i \leq b\}$ of task-PIOAs, where each A_i has description bounded by some $p_i \in \mathbb{N}$. It is not hard to check that the composition $\|_{i=1}^{b} \mathcal{A}_i$ has description bounded by $c_{\mathsf{comp}} \cdot \sum_{i=1}^{b} p_i$, where c_{comp} is a fixed constant. (The proof of this result in an immediate extension of the binary case described in [20, Lemma 4.2]).

To reason about the hiding operator in a setting with time bounds, we need the notion of p-time recognizable sets. Given a set S of binary strings and $p \in \mathbb{N}$, we say that S is *p-time recognizable* if there is a probabilistic Turing machine M satisfying: (i) in time at most p, M decides if a binary string a is in the set S, and (ii) the description of M has at most p bits under some standard encoding. If $S \subseteq Act_{\mathcal{A}}$ for some PIOA $\mathcal{A}$, then we say that S is p-time recognizable if the set of binary representations of actions in S is p-time recognizable. We claim there exists a constant c_{hide} such that, for any task-PIOA with p-bounded description and any p'-time recognizable set S of output actions of $\mathcal{A}$, the task-PIOA $\mathsf{hide}(\mathcal{A}, S)$ has $c_{\mathsf{hide}}(p + p')$-bounded description [20, Lemma 4.4].

A *task-PIOA family* $\overline{\mathcal{A}}$ is an indexed set $\{\mathcal{A}_k\}_{k\in\mathbb{N}}$ of task-PIOAs. The index k is commonly referred to as the *security parameter*. We say that $\overline{\mathcal{A}}$ has *p-bounded description* for some $p : \mathbb{N} \to \mathbb{N}$ just in case: for all k, $\mathcal{A}_k$ has $p(k)$-bounded description. If p is a polynomial, then we say that $\overline{\mathcal{A}}$ has *polynomially-bounded description*. The notions of compatibility, parallel composition and hiding are defined pointwise. Time bound results for composition and hiding extend easily to the setting of families.

3 Approximate Implementation

In [21, 20], we propose an approximate implementation relation for task-PIOAs families, expressing the idea that every behavior of one family is computationally indistinguishable from some behavior of another family. Following a traditional approach in cryptography, this definition compares acceptance probabilities of a distinguisher environment that runs in parallel with the task-PIOAs in question. Moreover, it encapsulates additional quantification over schedule length bounds and the choices of task schedules. These types of quantification are new challenges, presented by the fact that we do not impose *a priori* bounds on schedule lengths (and hence on overall runtime) and that we allow nondeterministic specifications.

We shall first present the approximate implementation relation of [21, 20] and state a composition theorem for single substitution. Then we discuss the difficulties in generalizing to a polynomial number of substitutions. This leads to a new, stronger definition of approximate implementation, for which we prove a polynomial composition theorem.

3.1 The Weak Variant

We begin with the notions of acceptance probabilities and closing environment. Let $\mathcal{A}$ be a closed task-PIOA with a special output action acc and let ρ be a task schedule for $\mathcal{A}$. The *acceptance probability* of $\mathcal{A}$ under ρ is defined to be: $\mathbf{P}_{\mathsf{acc}}(\mathcal{A}, \rho) := \Pr[\beta \text{ contains } \mathsf{acc} : \beta \xleftarrow{\mathsf{R}} \mathsf{tdist}(\mathcal{A}, \rho)]$, that is, the probability that a trace drawn from the distribution $\mathsf{tdist}(\mathcal{A}, \rho)$ contains the action acc. Now suppose $\mathcal{A}$ is any task-PIOA, not necessarily closed. A task-PIOA *Env* is an *environment* for $\mathcal{A}$ if it is compatible with $\mathcal{A}$ and $\mathcal{A}\|Env$ is closed. Throughout this paper, we assume that every environment has acc as an output, so that we may speak of acceptance probabilities of $\mathcal{A}\|Env$.

Implementation relations are defined on task-PIOAs with the same external interface. More precisely, $\mathcal{A}_1$ and $\mathcal{A}_2$ are said to be *comparable* if $I_1 = I_2$ and $O_1 = O_2$. Observe that comparability implies $\mathcal{A}_1$ and $\mathcal{A}_2$ have the same set of environments, up to renaming of internal actions. Suppose $\mathcal{A}_1$ and $\mathcal{A}_2$ are indeed comparable. Let $\mathbb{R}^{\geq 0}$ denote the set of non-negative reals and let $\epsilon \in \mathbb{R}^{\geq 0}$ and $p, q_1, q_2 \in \mathbb{N}$ be given[3]. We define $\mathcal{A}_1 \leq_{p,q_1,q_2,\epsilon} \mathcal{A}_2$ as follows: given any

[3] As a convention, we use variable p for description bounds and variable q for schedule length bounds.

environment Env with p-bounded description and any q_1-bounded task schedule ρ_1 for $\mathcal{A}_1 \| Env$, there exists a q_2-bounded task schedule ρ_2 for $\mathcal{A}_2 \| Env$ such that $|\mathbf{P}_{\mathsf{acc}}(\mathcal{A}_1 \| Env, \rho_1) - \mathbf{P}_{\mathsf{acc}}(\mathcal{A}_2 \| Env, \rho_2)| \leq \epsilon$. In other words, from the perspective of an environment with p-bounded description, $\mathcal{A}_1$ and $\mathcal{A}_2$ "look almost the same" provided $\mathcal{A}_2 \| Env$ may take q_2 many steps whenever $\mathcal{A}_1 \| Env$ takes q_1 many steps.

The relation $\leq_{p,q_1,q_2,\epsilon}$ can be extended to task-PIOA families in the obvious way. Let $\overline{\mathcal{A}}_1 = \{(\mathcal{A}_1)_k\}_{k\in\mathbb{N}}$ and $\overline{\mathcal{A}}_2 = \{(\mathcal{A}_2)_k\}_{k\in\mathbb{N}}$ be (pointwise) comparable task-PIOA families. Given $\epsilon : \mathbb{N} \to \mathbb{R}^{\geq 0}$ and $p, q_1, q_2 : \mathbb{N} \to \mathbb{N}$, we say that $\overline{\mathcal{A}}_1 \leq_{p,q_1,q_2,\epsilon} \overline{\mathcal{A}}_2$ just in case $(\mathcal{A}_1)_k \leq_{p(k),q_1(k),q_2(k),\epsilon(k)} (\mathcal{A}_2)_k$ for every k.

Restricting our attention to negligible error and polynomial time bounds, we obtain the approximate implementation $\leq_{\mathsf{neg,pt}}$. Formally, a function $\epsilon : \mathbb{N} \to \mathbb{R}^{\geq 0}$ is said to be *negligible* if, for every constant $c \in \mathbb{N}$, there exists $k_0 \in \mathbb{N}$ such that $\epsilon(k) < \frac{1}{k^c}$ for all $k \geq k_0$. (That is, ϵ diminishes more quickly than the reciprocal of any polynomial.) We say that $\overline{\mathcal{A}}_1 \leq_{\mathsf{neg,pt}} \overline{\mathcal{A}}_2$ if: $\forall p, q_1\ \exists q_2, \epsilon\ \overline{\mathcal{A}}_1 \leq_{p,q_1,q_2,\epsilon} \overline{\mathcal{A}}_2$, where p, q_1, q_2 are polynomials and ϵ is a negligible function.

The following binary composition theorem for $\leq_{p,q_1,q_2,\epsilon}$ and $\leq_{\mathsf{neg,pt}}$ is proven in [20].

Theorem 1. *Let $\epsilon : \mathbb{N} \to \mathbb{R}^{\geq 0}$ and $p, p_3, q_1, q_2 : \mathbb{N} \to \mathbb{N}$ be given. Let $\overline{\mathcal{A}}_1, \overline{\mathcal{A}}_2$ and $\overline{\mathcal{A}}_3$ be task-PIOAs families satisfying: $\overline{\mathcal{A}}_1$ and $\overline{\mathcal{A}}_2$ are comparable, and $\overline{\mathcal{A}}_3$ has p_3-bounded description and is compatible with both $\overline{\mathcal{A}}_1$ and $\overline{\mathcal{A}}_2$. Then the following holds.*

(1) If $\overline{\mathcal{A}}_1 \leq_{c_{\mathsf{comp}}(p+p_3),q_1,q_2,\epsilon} \overline{\mathcal{A}}_2$, where c_{comp} is the constant factor associated with description bounds in parallel composition, then $\overline{\mathcal{A}}_1 \| \overline{\mathcal{A}}_3 \leq_{p,q_1,q_2,\epsilon} \overline{\mathcal{A}}_2 \| \overline{\mathcal{A}}_3$.
(2) If $\overline{\mathcal{A}}_1 \leq_{\mathsf{neg,pt}} \overline{\mathcal{A}}_2$ and p_3 is a polynomial, then $\overline{\mathcal{A}}_1 \| \overline{\mathcal{A}}_3 \leq_{\mathsf{neg,pt}} \overline{\mathcal{A}}_2 \| \overline{\mathcal{A}}_3$.

Observe that, by induction, Theorem 1 generalizes to any constant number of substitutions.

3.2 Towards Polynomial Composition

For cryptographic applications, it is desirable to generalize Theorem 1 even further, to any polynomial number of substitutions. We now identify and discuss a few issues associated with this generalization.

Let us first examine the logical structure of the definition of $\leq_{\mathsf{neg,pt}}$.

$$\begin{aligned}
\overline{\mathcal{A}}_1 \leq_{\mathsf{neg,pt}} \overline{\mathcal{A}}_2 &\Leftrightarrow \forall p, q_1\ \exists q_2, \epsilon \quad \overline{\mathcal{A}}_1 \leq_{p,q_1,q_2,\epsilon} \overline{\mathcal{A}}_2 \\
&\Leftrightarrow \forall p, q_1\ \exists q_2, \epsilon \quad \forall k, Env, \rho_1\ \exists \rho_2 \\
&\qquad |\mathbf{P}_{\mathsf{acc}}((\mathcal{A}_1)_k \| Env, \rho_1) - \mathbf{P}_{\mathsf{acc}}((\mathcal{A}_2)_k \| Env, \rho_2)| \leq \epsilon,
\end{aligned}$$

where p, q_1, q_2 are polynomials, ϵ is a negligible function, Env is an environment for $\mathcal{A}_1$ with $p(k)$-bounded description, ρ_1 is a $q_1(k)$-bounded task schedule for $\mathcal{A}_1 \| Env$, and ρ_2 is a $q_2(k)$-bounded task schedule for $\mathcal{A}_2 \| Env$.

The outermost quantifiers, $\forall p, q_1 \exists q_2, \epsilon$, capture computational requirements: p bounds the description of a distinguisher environment, q_1 bounds the total number of steps that can be executed by $\mathcal{A}_1$ and an environment, q_2 bounds the total number of steps that can be executed by $\mathcal{A}_2$ and the same environment, and ϵ bounds the difference in acceptance probabilities. Intuitively, ϵ represents the degree to which $\mathcal{A}_1$ and $\mathcal{A}_2$ are indistinguishable, and we want to allow ϵ to depend on the computation power of the distinguisher environment. Since task-PIOAs do not have *a priori* bounds on the number of execution steps, we need the quantification $\forall q_1 \exists q_2$ to determine the number of steps that can be taken by $\mathcal{A}_1 \| Env$ and $\mathcal{A}_2 \| Env$, respectively. Note that the computation power of the environment is bounded by $p \cdot q_1$, therefore we allow ϵ to depend on both p and q_1. Moreover, the schedule length bound q_2 may be larger than q_1, giving $\mathcal{A}_2$ some freedom to perform more internal steps.

The innermost quantifiers, $\forall \rho_1 \exists \rho_2$, deal with nondeterministic choices in $\mathcal{A}_1$ and $\mathcal{A}_2$. We require that every schedule for $\mathcal{A}_1 \| Env$ can be matched by some schedule for $\mathcal{A}_2 \| Env$. Here "matching" means the acceptance probabilities differ by at most ϵ.

We would like to obtain a polynomial composition theorem, which would roughly say the following: given a polynomial b and two sequences of task-PIOA families $\overline{\mathcal{A}}_1^1, \overline{\mathcal{A}}_1^2, \ldots$ and $\overline{\mathcal{A}}_2^1, \overline{\mathcal{A}}_2^2, \ldots$ with $\overline{\mathcal{A}}_1^i \leq_{\mathsf{neg,pt}} \overline{\mathcal{A}}_2^i$ for all i, the family $\widehat{\mathcal{A}}_1$ defined by $(\widehat{\mathcal{A}}_1)_k := (\overline{\mathcal{A}}_1^1)_k \| \ldots \| (\overline{\mathcal{A}}_1^{b(k)})_k$ is again related by $\leq_{\mathsf{neg,pt}}$ to the family $\widehat{\mathcal{A}}_2$ defined by $(\widehat{\mathcal{A}}_2)_k := (\overline{\mathcal{A}}_2^1)_k \| \ldots \| (\overline{\mathcal{A}}_2^{b(k)})_k$. Such a theorem is proven in [35], with the assumption that errors in acceptance probabilities are uniformly bounded; that is, the same ϵ applies to $\overline{\mathcal{A}}_1^i$ and $\overline{\mathcal{A}}_2^i$ all i. The proof uses a typical *hybrid argument*, where, for each security parameter k, a sequence of $b(k)+1$ hybrids are constructed. The 0-th hybrid is $(\widehat{\mathcal{A}}_1)_k$, and the $i+1$th hybrid is obtained from the i-th hybrid by replacing $(\overline{\mathcal{A}}_1^{i+1})_k$ with $(\overline{\mathcal{A}}_2^{i+1})_k$. It is then argued that, since the error between each successive pair of hybrids is at most $\epsilon(k)$, the error between the 0-th and $b(k)$-th hybrids is at most $b(k) \cdot \epsilon(k)$. This is sufficient because the $b(k)$-th hybrid is precisely $(\widehat{\mathcal{A}}_2)_k$ and the function $b \cdot \epsilon$ is negligible whenever ϵ is negligible and b is polynomial.

In our setting, such a hybrid argument is much more difficult to construct, due to the additional quantification over schedule length bounds and choices of task schedules. To ensure that ϵ is independent of i, the uniformity condition becomes: $\forall p, q_1 \ \exists q_2, \epsilon \ \forall i \quad \overline{\mathcal{A}}_1^i \leq_{p,q_1,q_2,\epsilon} \overline{\mathcal{A}}_2^i$. Unfortunately, this does not appear sufficient for the hybrid argument, because, in order to guarantee the same error bound ϵ at each consecutive pair of hybrids, we would have to invoke the uniformity condition with the same p and q_1. This cannot be achieved because we have a new schedule length bound q_2, which need not be the same as q_1.

To be more concrete, let us fix a security parameter k and consider, for example, the 0-th hybrid $(\overline{\mathcal{A}}_1^1)_k \| \ldots \| (\overline{\mathcal{A}}_1^{b(k)})_k$. Let Env denote $(\overline{\mathcal{A}}_1^2)_k \| \ldots \| (\overline{\mathcal{A}}_1^{b(k)})_k$. Suppose we apply the uniformity condition with some appropriate p and q_1, obtaining q_2 and ϵ such that every $q_1(k)$-bounded schedule for $(\overline{\mathcal{A}}_1^1)_k \| Env$ can

be matched by some $q_2(k)$-bounded schedule for $(\overline{\mathcal{A}}_2^1)_k \| Env$. Then, in order to do the next replacement (i.e., replacing $(\overline{\mathcal{A}}_1^2)_k$ with $(\overline{\mathcal{A}}_2^2)_k$), we would have to instantiate the uniformity condition with q_2, leading to a possibly different error bound ϵ'.

This suggests the outermost quantification $\forall p, q_1 \exists q_2, \epsilon$ in $\mathcal{A}_1 \leq_{\mathsf{neg,pt}} \mathcal{A}_2$ does not capture correctly the idea that $\mathcal{A}_1$ and $\mathcal{A}_2$ are indistinguishable by the *same* environment. Indeed, the schedule length bound $q_2(k)$ applies to the composite $(\overline{\mathcal{A}}_2^1)_k \| Env$, which may allow *Env* to take more steps than it does in the composite $(\overline{\mathcal{A}}_1^1)_k \| Env$.

These observations inspire several changes to strengthen the definition of $\leq_{\mathsf{neg,pt}}$. We would like to make sure that the new bound q_2 applies only to the newly substituted component and not to the environment, and that the choice of q_2 does not depend on the computation power of the environment. Moreover, we require that the tasks controlled by the environment are preserved at each substitution. These changes lead to a new approximate simulation relation, $\leq_{\mathsf{neg,pt}}^{\mathsf{strong}}$, for which the uniformity condition can be invoked with the same bounds at each step of the hybrid argument.

3.3 The Strong Variant

In order to implement the changes suggested above (in particular, to have separate schedule length bounds for the components and the environment), we need a notion of *projection* on task schedules. Suppose we have compatible task-PIOAs $\mathcal{A}_1$ and $\mathcal{A}_2$ with $\mathcal{A}_1 \| \mathcal{A}_2$ closed. Given a task schedule ρ for $\mathcal{A}_1 \| \mathcal{A}_2$, $\mathsf{proj}_1(\rho)$ is defined to be the restriction of ρ to tasks in $\mathcal{R}_1$. Similarly for $\mathsf{proj}_2(\rho)$.

Using this projection operator, we define a new implementation relation.

Definition 1. *Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be comparable task-PIOAs and let $\epsilon \in \mathbb{R}^{\geq 0}$ and $p, q, q_1, q_2 \in \mathbb{N}$ be given. We define $\mathcal{A}_1 \leq_{q_1,q_2,p,q,\epsilon} \mathcal{A}_2$ as follows: given any environment Env with p-bounded description and any task schedule ρ_1 for $\mathcal{A}_1 \| Env$ such that:*
- $\mathsf{proj}_{\mathcal{A}_1}(\rho_1)$ *is q_1-bounded, and*
- $\mathsf{proj}_{Env}(\rho_1)$ *is q-bounded,*

there is a task schedule ρ_2 for $\mathcal{A}_2 \| Env$ such that
- $\mathsf{proj}_{\mathcal{A}_2}(\rho_2)$ *is q_2-bounded,*
- $\mathsf{proj}_{Env}(\rho_1) = \mathsf{proj}_{Env}(\rho_2)$*, and*
- $|\,\mathbf{P}_{\mathsf{acc}}(\mathcal{A}_1 \| Env, \rho_1) - \mathbf{P}_{\mathsf{acc}}(\mathcal{A}_2 \| Env, \rho_2)| \leq \epsilon$.

This definition strengthens $\leq_{p,q_1,q_2,\epsilon}$ by requiring that the tasks controlled by *Env* are not affected by the substitution. Moreover, the schedule length bounds for the components and for the environment are considered separately, using projections of task schedules.

The relation $\leq_{q_1,q_2,p,q,\epsilon}$ can be extended to task-PIOA families in the same way as for $\leq_{p,q_1,q_2,\epsilon}$, and we claim that $\leq_{q_1,q_2,p,q,\epsilon}$ is transitive and preserved under hiding, with certain adjustments to errors and time bounds. Precise statements appear in Appendix A.

We use $\leq_{q_1,q_2,p,q,\epsilon}$ to define the strong approximate implementation relation.

Definition 2. *Suppose $\overline{\mathcal{A}}_1$ and $\overline{\mathcal{A}}_2$ are comparable task-PIOA families. We say that $\overline{\mathcal{A}}_1 \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \overline{\mathcal{A}}_2$ if $\forall q_1\ \exists q_2\ \forall p, q\ \exists \epsilon\quad \overline{\mathcal{A}}_1 \leq_{q_1,q_2,p,q,\epsilon} \overline{\mathcal{A}}_2$, where q_1, q_2, p, q are polynomials and ϵ is a negligible function.*

Notice that, unlike in the definition of $\leq_{\mathsf{neg,pt}}$, the schedule length bound q_2 for $\overline{\mathcal{A}}_2$ no longer depends on the environment bounds p and q. This is crucial for the hybrid argument in the composition proof for $\leq_{q_1,q_2,p,q,\epsilon}$ (Lemma 2). More precisely, because of this property, the same q_2 bound applies at each substitution, even though the schedule length bound of the environment may change due to previous substitutions[4].

We now proceed to prove the polynomial composition theorem for $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$. Lemma 1 gives a description bound for the composition of b task-PIOAs, assuming the description bounds of the individual task-PIOAs are bounded by a non-decreasing function.

Lemma 1. *Let $b \in \mathbb{N}$ and a sequence of task-PIOAs $\mathcal{A}^1, \mathcal{A}^2, \ldots, \mathcal{A}^b$ be given. Suppose there exists a non-decreasing function $r : \mathbb{N} \to \mathbb{N}$ such that, for all i, $\mathcal{A}^i$ has description bounded by $r(i)$. Then $\|_i \mathcal{A}^i$ has description bounded by $c_{\mathsf{comp}} \cdot b \cdot r(b)$, where c_{comp} is the constant factor for composing task-PIOAs in parallel.*

Proof. Since r is non-decreasing, we have $c_{\mathsf{comp}} \cdot \sum_{i=1}^{b} r(i) \leq c_{\mathsf{comp}} \cdot b \cdot r(b)$. □

Lemma 2 is essentially the hybrid argument in the polynomial composition theorem for $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$ (Theorem 2). It shows that $\leq_{q_1,q_2,p,q,\epsilon}$ is "preserved" under b-ary composition, provided the time bounds and errors are calibrated appropriately.

Lemma 2. *Let $b \in \mathbb{N}$ and two sequences of task-PIOAs*

$$\mathcal{A}_1^1, \mathcal{A}_1^2, \ldots, \mathcal{A}_1^b \text{ and } \mathcal{A}_2^1, \mathcal{A}_2^2, \ldots, \mathcal{A}_2^b$$

be given. Assume that, in each sequence, all task-PIOAs are pairwise compatible. Suppose there exist a non-decreasing function $r : \mathbb{N} \to \mathbb{N}$ such that, for all i, both $\mathcal{A}_1^i$ and $\mathcal{A}_2^i$ have description bounded by $r(i)$.

Let $q_1, q_2, q_2', p, p', q, q' \in \mathbb{N}$ and $\epsilon, \epsilon' \in \mathbb{R}^{\geq 0}$ be given. Assume the following.

(1) $p = c_{\mathsf{comp}} \cdot (b \cdot r(b) + p')$, where c_{comp} is the constant factor for composing task-PIOAs in parallel.

(2) $q_2' = q_1 + b \cdot q_2$; $q = q_1 + b \cdot q_2 + q'$; and $\epsilon' = b \cdot \epsilon$.

(3) For all i, $\mathcal{A}_1^i$ and $\mathcal{A}_2^i$ are comparable and $\mathcal{A}_1^i \leq_{q_1,q_2,p,q,\epsilon} \mathcal{A}_2^i$.

Then we have $\|_{i=1}^{b} \mathcal{A}_1^i \leq_{q_1,q_2',p',q',\epsilon'} \|_{i=1}^{b} \mathcal{A}_2^i$.

[4] Recall that, in a single step of the hybrid argument, the environment is the parallel composition of the original environment and all protocol instances that are not being replaced in the current step.

Before diving into the proof of Lemma 2, we take a moment to dissect the assumptions. First, we note that Assumption (3) is the uniformity condition, saying that the same time bounds and error can be used for every index i. To explain Assumptions (1) and (2), we need to briefly outline our proof strategy.

To prove $\|_{i=1}^{b}\mathcal{A}_1^i \leq_{q_1,q_2',p',q',\epsilon'} \|_{i=1}^{b}\mathcal{A}_2^i$, we take an environment Env for both $\|_{i=1}^{b}\mathcal{A}_1^i$ and $\|_{i=1}^{b}\mathcal{A}_2^i$. The description bound of Env is p'. In each step of the hybrid argument, we perform exactly one substitution, with all other components fixed. We may then view the composition of Env with all fixed components as an environment Env' for the component being substituted. The description of Env' is therefore bounded by $p = c_{\mathsf{comp}} \cdot (b \cdot r(b) + p')$, as in Assumption (1).

Now, q_1 is the schedule length bound for $\|_{i=1}^{b}\mathcal{A}_1^i$. Since we don't not know how the tasks are distributed among the b components, we use a conservative estimate: the schedule length bound for each $\mathcal{A}_1^i$ is also q_1, as in Assumption (3). Then, at each step of the hybrid argument, the schedule length increases by at most q_2, hence the schedule length bound for $\|_{i=1}^{b}\mathcal{A}_2^i$ is $q_2' = q_1 + b \cdot q_2$, as in Assumption (2). Similarly, the schedule length bound for Env' at each step of the hybrid argument must be at least $q = q_1 + b \cdot q_2 + q'$, as in Assumption (2). Finally, the errors accumulate at each step, so we multiply ϵ with a factor of b to obtain ϵ', as in Assumption (2).

Proof (Lemma 2). Let $\widehat{\mathcal{A}}_1$ and $\widehat{\mathcal{A}}_2$ denote $\|_i\mathcal{A}_1^i$ and $\|_i\mathcal{A}_2^i$, respectively. Unwinding the definition of $\leq_{q_1,q_2',p',q',\epsilon'}$, we need to prove: for every environment Env with p'-bounded description and task schedule ρ_1 for $\widehat{\mathcal{A}}_1\|Env$ such that

- $\mathsf{proj}_{\widehat{\mathcal{A}}_1}(\rho_1)$ is q_1-bounded, and
- $\mathsf{proj}_{Env}(\rho_1)$ is q'-bounded,

there exists task schedule ρ_2 for $\widehat{\mathcal{A}}_2\|Env$ such that

- $\mathsf{proj}_{\widehat{\mathcal{A}}_2}(\rho_2)$ is q_2'-bounded,
- $\mathsf{proj}_{Env}(\rho_1) = \mathsf{proj}_{Env}(\rho_2)$, and
- $|\,\mathbf{P}_{\mathsf{acc}}(\widehat{\mathcal{A}}_1\|Env, \rho_1) - \mathbf{P}_{\mathsf{acc}}(\widehat{\mathcal{A}}_2\|Env, \rho_2)| < \epsilon'$.

Let such Env and ρ_1 be given. For $1 \leq i \leq b-1$, let H^i denote the i-th hybrid automaton: $\mathcal{A}_2^1\| \ldots \|\mathcal{A}_2^i\|\mathcal{A}_1^{i+1}\| \ldots \|\mathcal{A}_1^b$.

Consider $i = 1$ and let $Env_1 := \mathcal{A}_1^2\| \ldots \|\mathcal{A}_1^b\|Env$. Clearly, Env_1 is an environment for both $\mathcal{A}_1^1$ and $\mathcal{A}_2^1$ and, by Lemma 1 and Assumption (1), Env_1 has p-bounded description. By the choice of ρ_1, we know that $\mathsf{proj}_{\mathcal{A}_1^1}(\rho_1)$ is q_1-bounded and $\mathsf{proj}_{Env_1}(\rho_1)$ is (q_1+q')-bounded. By Assumption (2), $\mathsf{proj}_{Env_1}(\rho_1)$ is q-bounded.

Now we apply Assumption (3) and choose task schedule ρ_2 for $H^1\|Env$ such that

- $\mathsf{proj}_{\mathcal{A}_2^1}(\rho_2)$ is q_2-bounded,
- $\mathsf{proj}_{Env_1}(\rho_1) = \mathsf{proj}_{Env_1}(\rho_2)$, and
- $|\,\mathbf{P}_{\mathsf{acc}}(\widehat{\mathcal{A}}_1\|Env, \rho_1) - \mathbf{P}_{\mathsf{acc}}(H^1\|Env, \rho_2)| < \epsilon$.

Note that, since $\mathcal{A}_1^2$ is part of Env_1, $\mathsf{proj}_{\mathcal{A}_1^2}(\rho_1) = \mathsf{proj}_{\mathcal{A}_1^2}(\rho_2)$. Therefore, $\mathsf{proj}_{\mathcal{A}_1^2}(\rho_2)$ is q_1-bounded. Similarly, $\mathsf{proj}_{H^1}(\rho_2)$ is (q_1+q_2)-bounded and ρ_2 is (q_1+q_2+q')-bounded.

Now consider $i = 2$ and let $Env_2 := \mathcal{A}_2^1 \| \mathcal{A}_1^3 \| \ldots \| \mathcal{A}_1^b \| Env$. As before, Env_2 is an environment for both $\mathcal{A}_1^2$ and $\mathcal{A}_2^2$, and it has p-bounded description. Moreover, $\mathsf{proj}_{\mathcal{A}_1^2}(\rho_2)$ is q_1-bounded and, since ρ_2 is $(q_1 + q_2 + q')$-bounded, $\mathsf{proj}_{Env_2}(\rho_2)$ is also $(q_1 + q_2 + q')$-bounded. By Assumption (2), $\mathsf{proj}_{Env_2}(\rho_2)$ is q-bounded.

Again, we apply Assumption (3) and choose task schedule ρ_3 for $H^2 \| Env$ such that

- $\mathsf{proj}_{\mathcal{A}_2^2}(\rho_3)$ is q_2-bounded,
- $\mathsf{proj}_{Env_2}(\rho_2) = \mathsf{proj}_{Env_2}(\rho_3)$, and
- $|\,\mathbf{P}_{\mathsf{acc}}(H^1 \| Env, \rho_2) - \mathbf{P}_{\mathsf{acc}}(H^2 \| Env, \rho_3)| < \epsilon$.

Note that, since $\mathcal{A}_1^3$ is part of both Env_1 and Env_2, we have $\mathsf{proj}_{\mathcal{A}_1^3}(\rho_3) = \mathsf{proj}_{\mathcal{A}_1^3}(\rho_2) = \mathsf{proj}_{\mathcal{A}_1^3}(\rho_1)$. Therefore, $\mathsf{proj}_{\mathcal{A}_1^3}(\rho_3)$ is q_1-bounded. Similarly, $\mathsf{proj}_{H^2}(\rho_3)$ is $(q_1 + 2 \cdot q_2)$-bounded and ρ_3 is $(q_1 + 2 \cdot q_2 + q')$-bounded.

Repeating the same argument for all hybrid automata, we obtain

$$
\begin{aligned}
&|\,\mathbf{P}_{\mathsf{acc}}(\widehat{\mathcal{A}}_1 \| Env, \rho_1) - \mathbf{P}_{\mathsf{acc}}(\widehat{\mathcal{A}}_2 \| Env, \rho_{b+1})| \\
&\leq |\,\mathbf{P}_{\mathsf{acc}}(\widehat{\mathcal{A}}_1 \| Env, \rho_1) - \mathbf{P}_{\mathsf{acc}}(H^1 \| Env, \rho_2)| \\
&\quad + |\,\mathbf{P}_{\mathsf{acc}}(H^1 \| Env, \rho_2) - \mathbf{P}_{\mathsf{acc}}(H^2 \| Env, \rho_3)| \\
&\quad + \ldots + |\,\mathbf{P}_{\mathsf{acc}}(H^{b-1} \| Env, \rho_b) - \mathbf{P}_{\mathsf{acc}}(\widehat{\mathcal{A}}_2 \| Env, \rho_{b+1})| \\
&< b \cdot \epsilon = \epsilon'
\end{aligned}
$$

Moreover, since Env is part of Env_i for every i, we know that $\mathsf{proj}_{Env}(\rho_{b+1}) = \mathsf{proj}_{Env}(\rho_1)$. Finally, we have that $\mathsf{proj}_{\widehat{\mathcal{A}}_2}(\rho_{b+1})$ is bounded by $q_2' = q_1 + b \cdot q_2$. This completes the proof that $\widehat{\mathcal{A}}_1 \leq_{q_1, q_2', p', q', \epsilon'} \widehat{\mathcal{A}}_2$. □

Theorem 2 now follows as a corollary of Lemma 2. Essentially, we expand the definition of $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$ and instantiate the time bounds and error with appropriate values.

Theorem 2 (Polynomial Composition Theorem for $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$). *Let two sequences of task-PIOA families $\overline{\mathcal{A}}_1^1, \overline{\mathcal{A}}_1^2, \ldots$ and $\overline{\mathcal{A}}_2^1, \overline{\mathcal{A}}_2^2, \ldots$ be given, with $\overline{\mathcal{A}}_1^i$ comparable to $\overline{\mathcal{A}}_2^i$ for all i. Assume further that, in each sequence, all task-PIOA families are pairwise compatible.*

Suppose there exist polynomials $r, s : \mathbb{N} \to \mathbb{N}$ such that, for all i, k, both $(\overline{\mathcal{A}}_1^i)_k$ and $(\overline{\mathcal{A}}_2^i)_k$ have description bounded by $r(i) \cdot s(k)$. Assume that r is non-decreasing. Assume further that

$$\forall q_1 \ \exists q_2 \ \forall p, q \ \exists \epsilon \ \forall i \ \overline{\mathcal{A}}_1^i \leq_{q_1, q_2, p, q, \epsilon} \overline{\mathcal{A}}_2^i, \tag{1}$$

where q_1, q_2, p, q are polynomials and ϵ is a negligible function. (This is a strengthening of the statement $\forall i \ \overline{\mathcal{A}}_1^i \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \overline{\mathcal{A}}_2^i$.)

Let b be any polynomial. For each k, let $(\widehat{\mathcal{A}}_1)_k$ denote $(\overline{\mathcal{A}}_1^1)_k \| \ldots \| (\overline{\mathcal{A}}_1^{b(k)})_k$. Similarly for $(\widehat{\mathcal{A}}_2)_k$. Then we have $\widehat{\mathcal{A}}_1 \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \widehat{\mathcal{A}}_2$.

Proof. By the definition of $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$, we need to prove:

$$\forall q_1' \ \exists q_2' \ \forall p', q' \ \exists \epsilon' \ \widehat{\mathcal{A}}_1 \leq_{q_1', q_2', p', q', \epsilon'} \widehat{\mathcal{A}}_2,$$

where q_1', q_2', p', q' are polynomials and ϵ' is a negligible function.

Let polynomial q_1' be given and set $q_1 := q_1'$. Choose q_2 according to Assumption (1) in the theorem statement. Set $q_2' := q_1 + b \cdot q_2$. Let polynomials p' and q' be given. Define:

(i) $p := c_{\mathsf{comp}} \cdot (p' + b \cdot (r \circ b))$, where c_{comp} is the constant factor for composing task-PIOAs in parallel;

(ii) $q := q_1 + b \cdot q_2 + q'$.

Now choose ϵ using q_1, q_2, p, q and Assumption (1), and define $\epsilon' := b \cdot \epsilon$.

Let $k \in \mathbb{N}$ be given. Observe that

- the task-PIOAs $(\overline{\mathcal{A}}_1^1)_k, \ldots, (\overline{\mathcal{A}}_1^{b(k)})_k, (\overline{\mathcal{A}}_2^1)_k, \ldots, (\overline{\mathcal{A}}_2^{b(k)})_k$,
- the function $s(k) \cdot r$ and
- the numbers $b(k), q_1(k), q_2(k), q_2'(k), p(k), p'(k), q(k), q'(k), \epsilon(k), \epsilon'(k)$

satisfy the assumptions in the statement of Lemma 2. Therefore we may conclude that $(\widehat{\mathcal{A}}_1)_k \leq_{q_1(k), q_2'(k), p'(k), q'(k), \epsilon'(k)} (\widehat{\mathcal{A}}_2)_k$. Since $q_1 = q_1'$, this completes the proof. □

To conclude this section, we obtain the constant composition theorem for $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$ (Corollary 1) as a corollary of Theorem 2. For this special case, we need not assume a uniformity condition, because we can consider maximum time bounds and maximum errors. We use the fact that $\leq_{q_1, q_2, p, q, \epsilon}$ is preserved if we relax the time bound q_2 and the error bound ϵ.

Lemma 3. *Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be comparable task-PIOAs and let $q_1, q_2, p, q \in \mathbb{N}$ and $\epsilon \in \mathbb{R}^{\geq 0}$ be given. Assume $\mathcal{A}_1 \leq_{q_1, q_2, p, q, \epsilon} \mathcal{A}_2$. For any $\hat{q}_2 \geq q_2$ and $\hat{\epsilon} \geq \epsilon$, we have $\mathcal{A}_1 \leq_{q_1, \hat{q}_2, p, q, \hat{\epsilon}} \mathcal{A}_2$.*

Corollary 1. *Let $B \in \mathbb{N}$ and two sequences of task-PIOA families $\overline{\mathcal{A}}_1^1, \overline{\mathcal{A}}_1^2, \ldots \overline{\mathcal{A}}_1^B$ and $\overline{\mathcal{A}}_2^1, \overline{\mathcal{A}}_2^2, \ldots, \overline{\mathcal{A}}_2^B$ be given, with $\overline{\mathcal{A}}_1^i$ comparable to $\overline{\mathcal{A}}_2^i$ for all i. Suppose there exists polynomial $s : \mathbb{N} \to \mathbb{N}$ such that, for all i, k, both $(\overline{\mathcal{A}}_1^i)_k$ and $(\overline{\mathcal{A}}_2^i)_k$ have description bounded by $s(k)$. Assume further that $\overline{\mathcal{A}}_1^i \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \overline{\mathcal{A}}_2^i$ for all $1 \leq i \leq B$.*

For each k, let $(\widehat{\mathcal{A}}_1)_k$ denote $(\overline{\mathcal{A}}_1^1)_k \| \ldots \| (\overline{\mathcal{A}}_1^B)_k$. Similarly for $(\widehat{\mathcal{A}}_2)_k$. Then we have $\widehat{\mathcal{A}}_1 \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \widehat{\mathcal{A}}_2$.

Proof. We claim that Assumption (1) in Theorem 2 is satisfied. Let polynomial q_1 be given. For each i, choose polynomial q_2^i using the assumption $\overline{\mathcal{A}}_1^i \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \overline{\mathcal{A}}_2^i$. Let $\hat{q}_2$ be any polynomial upperbound of $q_2^1, \ldots, q_2^B$.

Let polynomials p and q be given. For each i, choose negligible function ϵ^i using q_1, q_2^i, p, q and the assumption $\overline{\mathcal{A}}_1^i \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \overline{\mathcal{A}}_2^i$. Let $\hat{\epsilon}$ be $\max(\epsilon^1, \ldots, \epsilon^B)$.

Now we have $\overline{\mathcal{A}}_1^i \leq_{q_1, q_2^i, p, q, \epsilon^i} \overline{\mathcal{A}}_2^i$ for all i. By Lemma 3, this implies $\overline{\mathcal{A}}_1^i \leq_{q_1, \hat{q}_2, p, q, \hat{\epsilon}} \overline{\mathcal{A}}_2^i$ for all i, which is precisely Assumption (1) in Theorem 2.

Finally, let b be the constant polynomial $\underline{B}$ and let r be the constant polynomial $\underline{1}$. We apply Theorem 2 to conclude that $\widehat{\mathcal{A}}_1 \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \widehat{\mathcal{A}}_2$. □

4 Structures

In the previous sections, we defined and established properties of our model of concurrent computation, which is not specific to cryptographic protocols. On top of this "foundational layer", this section introduces our "security layer".

In the spirit of [17], we first define *structures*, which we use to specify protocols. To this purpose, we classify external actions of a task-PIOA into two categories: environment actions and adversary actions. Intuitively, environment actions are used to model the functional input/output interface of a protocol, whereas adversary actions are used to model network communications. This allows us to impose syntactic constraints on adversary task-PIOAs so that they do not have immediate access to protocol inputs and outputs.

Definition 3. *A* structure π *is a pair* $\langle \mathcal{A}, EAct \rangle$*, where* $\mathcal{A}$ *is a task-PIOA and* $EAct$ *is a subset of the external actions of* $\mathcal{A}$*, called the* environment actions*. The set of* adversary actions *is defined to be* $AAct := (I \cup O) \setminus EAct$*. For convenience, we also define: (i)* $EI := EAct \cap I$ *(*environment inputs*), (ii)* $EO := EAct \cap O$ *(*environment outputs*), (iii)* $AI := AAct \cap I$ *(*adversary inputs*) and (iv)* $AO = AAct \cap O$ *(*adversary inputs*).*

The notion of structure suggests the following definition of an adversary that may interact with a structure.

Definition 4. *An* adversary *for the structure* $\pi = \langle \mathcal{A}, EAct \rangle$ *is a task-PIOA* Adv *satisfying the following conditions: (i)* Adv *is compatible with* $\mathcal{A}$*, (ii)* $AI \subseteq Act_{Adv}$*, and (iii)* $Act_{Adv} \cap EAct = \emptyset$*.*

In other words, Adv is a compatible task-PIOA that interacts with π via adversary actions only, and Adv provides all adversary inputs to π.

Two structures π_1 and π_2 are said to be *comparable* if $EI_1 = EI_2$ and $EO_1 = EO_2$. Notice, unlike comparability for task-PIOAs, comparability for structures ignores differences in adversary actions.

Two structures π_1 and π_2 are *compatible* if $\mathcal{A}_1$ and $\mathcal{A}_2$ are compatible task-PIOAs and $Act_1 \cap Act_2 = EAct_1 \cap EAct_2$. That is, every shared action must be an environment action of both structures. Composition is straightforward: given compatible π_1 and π_2, their *composition* $\pi_1 \| \pi_2$ is the structure $\langle \mathcal{A}_1 \| \mathcal{A}_2, EAct_1 \cup EAct_2 \rangle$. This definition can be extended to any finite number of components. We observe that an adversary for a composition of structures is also an adversary for each of the component structures. A proof of this result is given in Appendix C.

Finally, we consider hiding for structures. Given a structure $\langle \mathcal{A}, EAct \rangle$ and a set S of output actions of $\mathcal{A}$, we define $hide(\langle \mathcal{A}, EAct \rangle, S)$ to be the structure $\langle hide(\mathcal{A}, S), EAct \setminus S \rangle$.

Time Bounds A structure $\pi = (\mathcal{A}, EAct)$ is said to have *p-bounded* description if $\mathcal{A}$ has p-bounded description and $EAct$ is p-time recognizable. We observe that the composition of bounded structures has a description bound linear in the sum of component bounds. Similarly, the hiding operator increases the description bound by a fixed constant factor. More details about these results are available in Appendix B.

Structure Families Given a family $\bar{\pi}$ of structures and a function $p : \mathbb{N} \to \mathbb{N}$, we say that $\bar{\pi}$ has *p-bounded* description if π_k has $p(k)$-bounded description for every k. If p is a polynomial, then we say that $\bar{\pi}$ has *polynomially-bounded* description.

The notions of comparability, compatibility and parallel composition are defined pointwise. Similarly for the notion of an adversary family.

If $\bar{S} = \{S_k\}_{k\in\mathbb{N}}$ is a family of sets of actions, we say that $\bar{S}$ is *polynomial-time recognizable* if there is a polynomial p such that every S_k is $p(k)$-time recognizable. It is not hard to check that, given any family $\bar{\pi}$ with polynomially-bounded description and a polynomial-time recognizable family $\bar{S}$ of sets of actions, the family $\mathsf{hide}(\bar{\pi}, \bar{S})$ is again polynomial time-bounded. Those results are detailed in Appendix B.

5 Secure Emulation

Equipped with the notions of polynomial-time-bounded structure and adversary families, we have now enough machinery to formulate our secure emulation notion. To this purpose, we follow the standard definition of universal composability/simulatability [19, 17].

Definition 5 (Secure Emulation). *Suppose ϕ and ψ are comparable structure families. We say that ϕ* emulates *ψ (denoted $\phi \leq_{\mathsf{SE}} \psi$) if, for every adversary family Adv for ϕ with polynomially bounded description, there is an adversary family Sim for ψ with polynomially bounded description such that:*

$$\mathsf{hide}(\phi \| Adv, AAct_\phi) \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \mathsf{hide}(\psi \| Sim, AAct_\psi).$$

Transitivity of $\leq_{\mathsf{SE}}$ follows immediately from transitivity of $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$.

Dummy Adversaries Observe that, in the definition of $\leq_{\mathsf{SE}}$, the adversary actions of ϕ and ψ are hidden, which prevents an environment from synchronizing on those actions. At first sight, this limits the amount of information available to the environment and hence reduces its distinguishing power. However, one can show that no power is actually lost, because there exist adversaries that behave simply as forwarders between *Env* and the protocols. These are the so-called *dummy adversaries* and below we give a canonical construction.

Let ϕ be a structure family and, for each $k \in \mathbb{N}$, let f_k be a bijection from $AAct_{\phi_k}$ to a set of fresh action names. We refer to $f = \{f_k\}_{k\in\mathbb{N}}$ as a *renaming* of adversary actions for ϕ, and we write $f(\phi)$ for the result of applying f_k to ϕ_k for every k. Consider the adversary $Adv(\phi_k, f_k)$ defined in Figure 1.

The following lemma shows that dummy adversaries have transparent behavior. This fact is used to in the proof of our main composition theorem (Theorem 3).

Lemma 4. *Let $Adv(\phi, f)$ denote the family $\{Adv(\phi_k, f_k)\}_{k\in\mathbb{N}}$. Note that $f(\phi)$ and $\mathsf{hide}(\phi \| Adv(\phi, f), AAct_\phi)$ are comparable. Let A be a task-PIOA family compatible with both $f(\phi)$ and $\mathsf{hide}(\phi \| Adv(\phi, f), AAct_\phi)$. Assume that, for all k, $f(AI_{\phi_k}) \subseteq Act_{A_k}$. Then $f(\phi) \| A \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \mathsf{hide}(\phi \| Adv(\phi, f), AAct_\phi) \| A$.*

$Adv(\phi_k, f_k)$

Signature

Input:
$AO_{\phi_k} \cup f_k(AI_{\phi_k})$
Output:
$f_k(AO_{\phi_k}) \cup AI_{\phi_k}$

Tasks

forward $:= f_k(AO_{\phi_k}) \cup AI_{\phi_k}$

States

pending $\in AO_{\phi_k} \cup f_k(AI_{\phi_k}) \cup \bot$, initially $\bot$

Transitions:

$a \in AO_{\phi_k} \cup f_k(AI_{\phi_k})$
Effect:
$pending := a$

$b \in f_k(AO_{\phi_k})$
Precondition:
$b = f_k(pending)$
Effect:
$pending := \bot$

$b \in AI_{\phi_k}$
Precondition:
$f_k(b) = pending$
Effect:
$pending := \bot$

Fig. 1. Task-PIOA Code for Dummy Adversary

Proof. Let q_1 be any polynomial and set $q_2 := 2q_1$. Let p, q be any polynomials and ϵ be the constant polynomial $\underline{0}$. Fix $k \in \mathbb{N}$ and let Env be an environment for $f_k(\phi_k)\|A_k$ and for $\phi_k\|Adv(\phi_k, f_k)\|A_k$. Let ρ be a task schedule for $f_k(\phi_k)\|\mathcal{A}_k\|Env$ such that $\mathsf{proj}_{f_k(\phi_k)\|A_k}(\rho)$ is $q_1(k)$-bounded and $\mathsf{proj}_{Env}(\rho)$ is $q_(k)$-bounded.

We construct a task schedule ρ' for $\phi_k\|Adv(\phi_k, f_k)\|A_k\|Env$ as follows: given any task T that is not locally controlled by Env, we replace T with T.forward. Note that, by construction, $\mathsf{proj}_{\mathsf{hide}(\phi_k\|Adv(\phi_k,f_k),AAct_{\phi_k})}(\rho')$ is $q_2(k)$-bounded and $\mathsf{proj}_{Env}(\rho) = \mathsf{proj}_{Env}(\rho')$. Moreover, we have by assumption that $f(AI_{\phi_k}) \subseteq Act_{A_k}$, hence we have sufficiently many forward tasks to guarantee

$$\mathbf{P}_{\mathsf{acc}}(f_k(\phi_k)\|A_k\|Env, \rho) = \mathbf{P}_{\mathsf{acc}}(\mathsf{hide}(\phi_k\|Adv(\phi_k, f_k), AAct_{\phi_k})\|A_k\|Env, \rho').$$

□

Composition We now prove that $\leq_{\mathsf{SE}}$ is preserved under polynomial-sized composition, provided certain uniformity assumptions are satisfied.

Theorem 3. *Let two sequences of pairwise compatible structure families $\phi^1, \phi^2, \ldots$ and $\psi^1, \psi^2, \ldots$ be given, with ϕ^i comparable to ψ^i for all i.*

Suppose there are renamings $f^1, f^2, \ldots$ and polynomials $r, s : \mathbb{N} \to \mathbb{N}$ such that the following hold.
(1) r is non-decreasing.

(2) For all i, $\phi^i \| Adv(\phi^i, f^i)$ has description bounded by $r(i) \cdot s$. (The family $Adv(\phi^i, f^i)$ is a dummy adversary family, as in Lemma 4.)
(3) There exist adversary families $Sim^1, Sim^2, \ldots$ for $\psi^1, \psi^2, \ldots$ such that
(a) for all i, $\psi^i \| Sim^i$ has description bounded by $r(i) \cdot s$, and
(b) $\forall q_1\ \exists q_2\ \forall p, q\ \exists \epsilon\ \forall i$
$\mathsf{hide}(\phi^i \| Adv(\phi^i, f^i), AAct_{\phi^i}) \leq_{q_1,q_2,p,q,\epsilon} \mathsf{hide}(\psi^i \| Sim^i, AAct_{\psi^i})$,
where q_1, q_2, p, q are polynomials and ϵ is a negligible function.

Let b be any polynomial. For each k, let $\widehat{\phi}_k$ denote $\phi_k^1 \| \ldots \| \phi_k^{b(k)}$. Similarly for $\widehat{\psi}_k$. Then we have $\widehat{\phi} \leq_{\mathsf{SE}} \widehat{\psi}$.

Proof. Let Adv be an adversary family for $\widehat{\phi}$ with polynomially bounded description. We need to construct an adversary family Sim for $\widehat{\psi}$ with polynomially bounded description such that:

$$\mathsf{hide}(\widehat{\phi} \| Adv, AAct_{\widehat{\phi}}) \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \mathsf{hide}(\widehat{\psi} \| Sim, AAct_{\widehat{\psi}}).$$

Observe that the renamings $f^1, f^2, \ldots$ induce a renaming for $\widehat{\phi}$ in the obvious way: for each k, $f_k := f_k^1 \cup \ldots \cup f_k^{b(k)}$. This is well defined because the compatibility definition for structures requires the sets of adversary actions to be pairwise disjoint.

Let $\widehat{Adv}$ and $\widehat{Sim}$ be adversary families defined as follows: for each k,

$$\widehat{Adv}_k := Adv(\phi_k^1, f_k^1) \| \ldots \| Adv(\phi_k^{b(k)}, f_k^{b(k)}), \text{ and}$$

$$\widehat{Sim}_k := Sim_k^1 \| \ldots \| Sim_k^{b(k)},$$

where $Sim^1, Sim^2, \ldots$ are given as in the statement of the theorem.

We observe the following.

$$\begin{aligned}
&\mathsf{hide}(\widehat{\phi} \| Adv, AAct_{\widehat{\phi}}) \\
&\equiv_{\mathsf{neg,pt}} \mathsf{hide}(f(\widehat{\phi}) \| f(Adv), f(AAct_{\widehat{\phi}})) && \text{property of renaming} \\
&\leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \mathsf{hide}(\widehat{\phi} \| \widehat{Adv} \| f(Adv), f(AAct_{\widehat{\phi}}) \cup AAct_{\widehat{\phi}}) && \text{Lemma 4} \\
&\leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \mathsf{hide}(\widehat{\psi} \| \widehat{Sim} \| f(Adv), f(AAct_{\widehat{\phi}}) \cup AAct_{\widehat{\psi}}) && \text{Theorem 2} \\
&\equiv_{\mathsf{neg,pt}} \mathsf{hide}(\widehat{\psi} \| \mathsf{hide}(\widehat{Sim} \| f(Adv), f(AAct_{\widehat{\phi}})), AAct_{\widehat{\psi}}) && \text{property of hiding}
\end{aligned}$$

Diagrams depicting these task-PIOAs and the communications between them are in Figure 2. We define Sim to be $\mathsf{hide}(\widehat{Sim} \| f(Adv), f(AAct_{\widehat{\phi}}))$. This completes the proof. □

Let us now compare the assumptions of Theorem 3 with the more intuitive assumption that $\phi^i \leq_{\mathsf{SE}} \psi^i$ for all i. The latter is not sufficient for two reasons.

– We need to ensure that the composites $\widehat{\phi}$ and $\widehat{\psi}$ have polynomially bounded description. The same applies to the adversary families $\widehat{Adv}$ and $\widehat{Sim}$. Therefore we need the existence of polynomial bounds r and s. Note that we do

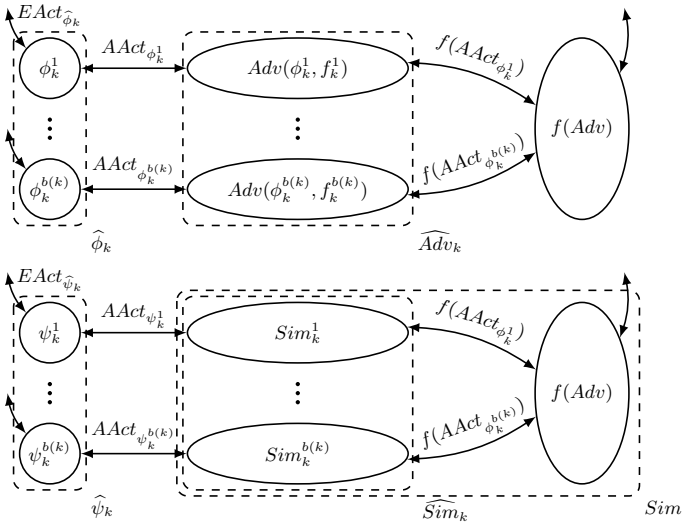

Fig. 2. Diagram for the Construction of *Sim*

allow the complexity to grow with i, as long as the growth in i is independent of the growth in the security parameter k. This is important because our compatibility condition requires disjoint sets of locally controlled actions: as i grows, action names need to contain more bits. (This can be thought of as the need to have distinct session IDs for different protocol instances.)

– Assumption (3b) is the so-called uniformity condition on the error in simulation. We require that the same error bound ϵ works for all instances i. This prevents the errors from growing with i, otherwise the total error may no longer be negligible.

We present two examples of applications of this composition theorem further in this section.

Hiding Our secure emulation relation is preserved when we hide any set of environment output actions of the related structures. This result can be naturally used to model the behavior of protocols privately synchronizing with subprotocols.

Theorem 4. *Suppose ϕ and ψ are comparable structure families such that $\phi \leq_{\mathsf{SE}} \psi$. Suppose also that $B \subset EO_\phi$ is a family of sets of environment output actions of ϕ. Then, $hide(\phi, B) \leq_{\mathsf{SE}} hide(\psi, B)$.*

Proof. Suppose ϕ, ψ and B are defined as in the hypotheses. Unwinding the assumption that $\phi \leq_{\mathsf{SE}} \psi$, we obtain that, for every polynomial time-bounded adversary family Adv for ϕ, there is a polynomial time-bounded adversary family Sim for ψ such that $hide(\phi \| Adv, AAct_\phi) \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} hide(\psi \| Sim, AAct_\psi)$.

Using the definition of adversaries, we observe that $Act_{Adv} \cap B = Act_{Sim} \cap B = \emptyset$. This guarantees that Adv is an adversary family for $hide(\phi, B)$ and that Sim is an adversary family for $hide(\psi, B)$. Now, using the hiding property of $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$, we obtain that $hide(hide(\phi \| Adv, AAct_{\phi}), B) \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} hide(hide(\psi \| Sim, AAct_{\psi}), B)$. Using the set intersection relations above and the fact that we only hide environment external actions, this implies $hide(hide(\phi, B) \| Adv, AAct_{hide(\phi,B)}) \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} hide(hide(\psi, B) \| Sim, AAct_{hide(\psi,B)})$, as needed. □

Applications We state two simple corollaries illustrating the use of our composition theorem: the first one considers composition for a polynomial number of copies of a single structure family, while the second considers composition for a constant number of distinct structure families. Proofs for these corollaries appear in Appendix D.

Corollary 2. *Suppose ϕ and ψ are comparable polynomial-time-bounded structure families such that $\phi \leq_{\mathsf{SE}} \psi$. Let $g^1, g^2, \ldots$ be renaming functions, each mapping actions of ϕ and ψ to fresh names. Suppose further that applying the renaming g^i to the family ϕ or ψ does not increase their time-bounds more than by a polynomial factor in the index i.*

Let b be a polynomial. For each k, let $\widehat{\phi}_k$ denote $g^1(\phi_k) \| \ldots \| g^{b(k)}(\phi_k)$, and similarly for $\widehat{\psi}_k$. Then we have $\widehat{\phi} \leq_{\mathsf{SE}} \widehat{\psi}$.

Corollary 3. *Let $\phi^1, \ldots, \phi^B$ and $\phi^1, \ldots, \psi^B$ be pairwise compatible polynomial-time-bounded structure families, with $\phi^i \leq_{\mathsf{SE}} \psi^i$ for every i. Then, we have $\phi^1 \| \cdots \| \phi^B \leq_{\mathsf{SE}} \psi^1 \| \cdots \| \psi^B$.*

6 Conclusions

In this paper, we introduced a new approximate implementation relation for task-PIOAs, the $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$ relation, and showed that it supports composition theorems for polynomially growing task-PIOA families. Building upon this $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$ relation, we presented a secure emulation relation, following the logical statement of universal composability/simulatability, and proved this relation is transitive and preserved under hiding. It also supports composition theorems for polynomially growing structure families. These three properties, as well as the invariant assertion and simulation relation techniques developed in [37, 21], are essential for the scalability of protocol analysis.

In future works, we would like to consider dynamic creation definitions for task-PIOAs: this would allow us to model environments (or structures) that can dynamically create new protocol instances at run time, as it is performed through the dynamic ITM invocation mechanism in the UC framework or through the bang operator "!" in the IITM framework. We believe such an enrichment to our framework would allow us to prove a stronger claim about the existence of simulators. Namely, there is a single simulator that can simulate b many protocol instances for any polynomial b.

We would also like to apply the model and methods we developed here to analyze security protocols that have not yet been the subject of much formal study, such as timing-based and long-lived security protocols, where our separation between the bounds on description and schedulers seems especially meaningful.

References

1. Dolev, D., Yao, A.C.: On the security of public-key protocols. IEEE Transactions on information theory **2**(29) (1983) 198–208
2. Abadi, M., Gordon, A.: A calculus for cryptographic protocols: the spi calculus. Information and Computation **148** (1999) 1–70
3. Burrows, M., Abadi, M., Needham, R.: A logic of authentication. ACM Transactions on Computer Systems **8**(1) (1990) 18–36
4. Durgin, N., Mitchell, J.C., Pavlovic, D.: A compositional logic for proving security properties of protocols. Journal of Computer Security **11**(4) (2003) 677–721
5. Lowe, G.: Some new attacks upon security protocols. In: Proceedings of 9th IEEE Computer Security Foundations Workshop, IEEE Computer Society Press (1996) 162–169
6. Lynch, N.: I/O automaton models and proofs for shared-key communication systems. In: 12th IEEE Computer Security Foundations Workshop — CSFW'99, Mordano, Italy, IEEE Computer Society Press (1999) 14–29
7. Meadows, C.: The NRL protocol analyzer : an overview. Journal of Logic Programming **26**(2) (1996) 113–131
8. Millen, J., Shmatikov, V.: Constraint solving for bounded-process cryptographic protocol analysis. In Samarati, P., ed.: Proceedings of 8th ACM Conference on Computer and Communications Security (CCS), ACM (2001) 166–175
9. Thayer, F.J., Herzog, J.H., Guttman, J.: Strand spaces: Proving security protocols correct. Journal of Computer Security **7**(2/3) (1999) 191–230
10. Abadi, M., Rogaway, P.: Reconciling two views of cryptography. In van Leeuwen, J., Watanabe, O., Hagiya, M., Mosses, P.D., Ito, T., eds.: Proceedings of the IFIP International Conference on Theoretical Computer Science 2000, Sendai, Japan, Springer-Verlag - LNCS Vol. 1872 (2000) 3–22
11. Canetti, R., Herzog, J.: Universally composable symbolic analysis of mutual authentication and key exchange protocols. In Halevi, S., Rabin, T., eds.: Proceedings, Theory of Cryptography Conference (TCC). Volume 3876 of LNCS., Springer (2006) 380–403 Full version available on `http://eprint.iacr.org/2004/334`.
12. Micciancio, D., Warinschi, B.: Soundness of formal encryption in the presence of active adversaries. In: Proceedings of the First Theory of Cryptography Conference, Cambridge, MA, USA, Springer-Verlag - LNCS Vol. 2951 (2004) 133–151
13. Lincoln, P., Mitchell, J., Mitchell, M., Scedrov, A.: A probabilistic poly-time framework for protocol analysis. In: Proceedings of the 5th ACM conference on Computer and communications security (CCS-5), San Francisco (1998) 112–121
14. Mateus, P., Mitchell, J., Scedrov, A.: Composition of cryptographic protocols in a probabilistic polynomial-time calculus. In Amadio, R., Lugiez, D., eds.: Proceedings of CONCUR 2003 - Concurrency Theory. Volume 2761 of LNCS., Marseille, France, Springer (2003) 327–349
15. Mitchell, J., Ramanathan, A., Scedrov, A., Teague, V.: A probabilistic polynomial-time process calculus for the analysis of cryptographic protocols. Theoretical Computer Science **353** (2006) 118–164

16. Pfitzmann, B., Waidner, M.: Composition and integrity preservation of secure reactive systems. In: Proc. of the 7th ACM Conference on Computer and Communications Security. (2000) 245–254
17. Pfitzmann, B., Waidner, M.: A model for asynchronous reactive systems and its application to secure message transmission. In: IEEE Symposium on Security and Privacy, Oakland, CA, IEEE Computer Society (2001) 184–200
18. Backes, M., Pfitzmann, B., Waidner, M.: Secure asynchronous reactive systems. Cryptology ePrint Archive, Report 2004/082 (2004) http://eprint.iacr.org/.
19. Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In Naor, M., ed.: Proceedings of the 42nd Annual Symposium on Foundations of Computer Science, IEEE Computer Society (2001) 136–145 Full version available on http://eprint.iacr.org/2000/067.
20. Canetti, R., Cheung, L., Kaynar, D., Liskov, M., Lynch, N., Pereira, O., Segala, R.: Using task-structured probabilistic I/O automata to analyze an oblivious transfer protocol. Cryptology ePrint Archive, Report 2005/452 (2005) http://eprint.iacr.org/.
21. Canetti, R., Cheung, L., Kaynar, D., Liskov, M., Lynch, N., Pereira, O., Segala, R.: Time-bounded Task-PIOAs: A framework for analyzing security protocols. In: Proceedings the 20th International Symposium on Distributed Computing (DISC 2006). (2006)
22. Küsters, R.: Simulation-Based Security with Inexhaustible Interactive Turing Machines. In: Proceedings of the 19th IEEE Computer Security Foundations Workshop (CSFW-19 2006), IEEE Computer Society (2006) 309–320
23. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof systems. In: Proceedings of the 17th Annual ACM Symposium on Theory of Computing (STOC'85). (1985) 291–304
24. Goldreich, O., Micali, S., Wigderson, A.: How to play any mental game a completeness theorem for protocols with honest majority. In: Proceedings of the 19th Annual ACM Symposium on the Theory of Computing (STOC), ACM Press (1987) 218–229
25. Goldwasser, S., Levin, L.: Fair computation of general functions in presence of immoral majority. In Menezes, A.J., Vanstone, S.A., eds.: Advances in Cryptology - Crypto '90, Berlin, Springer-Verlag (1990) 77–93 Lecture Notes in Computer Science Volume 537.
26. Beaver, D.: Secure multiparty protocols and zero-knowledge proof systems tolerating a faulty minority. Journal of Cryptology **4**(2) (1991) 75–122
27. Micali, S., Rogaway, P.: Secure computation. In Feigenbaum, J., ed.: Advances in Cryptology - Crypto '91, Berlin, Springer-Verlag (1991) 392–404 Lecture Notes in Computer Science Volume 576.
28. Pfitzmann, B., Waidner, M.: A general framework for formal notions of "secure" system. Technical report, Hildesheimer Informatik-Berichte 11/94, Institut fr Informatik, Universitt Hildesheim. (April 1994)
29. Canetti, R.: Studies in Secure Multi-Party Computation and Applications. PhD thesis, Weizmann Institute, Israel (1995)
30. Canetti, R., Cheung, L., Kaynar, D., Liskov, M., Lynch, N., Pereira, O., Segala, R.: Task-structured probabilistic I/O automata. Technical Report MIT-CSAIL-TR-2006-060, MIT CSAIL (2006) Submitted for journal publication. Up-to-date version available at http://theory.csail.mit.edu/~lcheung/papers/task-PIOA-TR.pdf.

31. Backes, M., Pfitzmann, B., Waidner, M.: A universally composable cryptographic library. Cryptology ePrint Archive, Report 2003/015 (2003) `http://eprint.iacr.org/`.
32. Datta, A., Kuesters, R., Mitchell, J.C., Ramanathan, A.: On the relationships between notions of simulation-based security. In Kilian, J., ed.: Proceedings of Theory of Cryptography Conference. Volume 3378 of LNCS., Springer (2005) 476–494 Full version available on `http://eprint.iacr.org/2006/153`.
33. Canetti, R., Cheung, L., Lynch, N., Pereira, O.: On the role of scheduling in simulation-based security. In: 7th International Workshop on Issues in the Theory of Security (WITS'07). (2006) To appear.
34. Mller-Quade, J., Unruh, D.: Long-term security and universal composability. In: Theory of Cryptography, Proceedings of TCC 2007. Lecture Notes in Computer Science, Springer-Verlag (2007) Preprint on IACR ePrint 2006/422, to appear.
35. Backes, M., Pfitzmann, B., Waidner, M.: A general composition theorem for secure reactive systems. In Naor, M., ed.: First Theory of Cryptography Conference, TCC 2004. Volume 2951 of LNCS., Cambridge, MA, USA, Springer-Verlag (2004) 336–354
36. Goldwasser, S., Micali, S.: Probabilistic encryption. Journal of Computer and Systems Sciences **28**(2) (1984) 270–299
37. Canetti, R., Cheung, L., Kaynar, D., Liskov, M., Lynch, N., Pereira, O., Segala, R.: Task-structured Probabilistic I/O Automata. In: Proceedings of the 8th International Workshop on Discrete Event Systems – WODES'2006. (2006) IEEE catalog number 06EX1259.
38. Lynch, N., Tuttle, M.: An introduction to input/output automata. CWI Quarterly **2**(3) (1989) 219–246
39. Segala, R., Lynch, N.: Probabilistic simulations for probabilistic processes. Nordic Journal of Computing **2**(2) (1995) 250–273
40. Datta, A., Kuesters, R., Mitchell, J.C., Ramanathan, A., Shmatikov, V.: Unifying equivalence-based definitions of protocol security. In: Proceedings of ACM SIGPLAN and IFIP WG 1.7 4th Workshop on Issues in the Theory of Security. (2004)

A Results for Task-PIOAs

We state the transitivity of the $\leq_{q_1,q_2,p,q,\epsilon}$ and $\leq^{\mathsf{strong}}_{\mathsf{neg,pt}}$ relations, and claim these relations are preserved when output actions of the related automata are hidden.

Lemma 5. *Suppose $\mathcal{A}_1$, $\mathcal{A}_2$ and $\mathcal{A}_3$ are comparable task-PIOAs such that $\mathcal{A}_1 \leq_{q_1,q_2,p,q,\epsilon_{12}} \mathcal{A}_2$ and $\mathcal{A}_2 \leq_{q_2,q_3,p,q,\epsilon_{23}} \mathcal{A}_3$. Then, $\mathcal{A}_1 \leq_{q_1,q_3,p,q,\epsilon_{13}+\epsilon_{23}} \mathcal{A}_3$.*

Lemma 6. *Suppose $\overline{\mathcal{A}_1} = \{(\mathcal{A}_1)_k\}_{k\in\mathbb{N}}$, $\overline{\mathcal{A}_2} = \{(\mathcal{A}_2)_k\}_{k\in\mathbb{N}}$ and $\overline{\mathcal{A}_3} = \{(\mathcal{A}_3)_k\}_{k\in\mathbb{N}}$ are comparable task-PIOA families such that $\overline{\mathcal{A}_1} \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \overline{\mathcal{A}_2}$ and $\overline{\mathcal{A}_2} \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \overline{\mathcal{A}_3}$. Then, $\overline{\mathcal{A}_1} \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \overline{\mathcal{A}_3}$.*

Lemma 7. *Suppose $\mathcal{A}_1$ and $\mathcal{A}_2$ are comparable task-PIOA families such that $\mathcal{A}_1 \leq_{q_1,q_2,p,q,\epsilon} \mathcal{A}_2$. Suppose also that B is set of output actions of both $\mathcal{A}_1$ and $\mathcal{A}_2$. Then, $hide(\mathcal{A}_1, B) \leq_{q_1,q_2,p,q,\epsilon} hide(\mathcal{A}_2, B)$.*

Lemma 8. *Suppose* $\overline{\mathcal{A}_1} = \{(\mathcal{A}_1)_k\}_{k\in\mathbb{N}}$ *and* $\overline{\mathcal{A}_2} = \{(\mathcal{A}_2)_k\}_{k\in\mathbb{N}}$ *are comparable task-PIOA families such that* $\overline{\mathcal{A}_1} \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} \overline{\mathcal{A}_2}$. *Suppose also that* $\overline{B} = \{B_k\}_{k\in\mathbb{N}}$ *is a family of sets of output actions of* $\overline{\mathcal{A}_1}$ *and* $\overline{\mathcal{A}_2}$, *that is,* B_k *is a set of output actions of both* $(\mathcal{A}_1)_k$ *and* $(\mathcal{A}_2)_k$. *Then,* $hide(\overline{\mathcal{A}_1}, \overline{B}) \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} hide(\overline{\mathcal{A}_2}, \overline{B})$.

The proof of these lemmas are similar to those appearing as [20, Lemma 4.9, 4.31, 4.11, and 4.33].

B Results for Structures

We consider the behavior of structures when they are composed.

Lemma 9. *There exists a constant* c_{comp} *such that the following holds. Suppose* $\pi_1, \pi_2, \ldots, \pi_n$ *are compatible structures, where, for every* $1 \leq i \leq n$, *the structure* π_i *is* b_i*-time bounded. Then,* $\pi_1 \| \cdots \| \pi_n$ *is* $c_{\mathsf{comp}}(b_1 + \cdots + b_n)$*-bounded. Also, the composition of* n *polynomial time-bounded structures is also a polynomial time-bounded structure.*

Proof. Similar to the proofs of [20, Lemma 4.2 and 4.26].

Corollary 4. *Suppose* $\overline{\pi} = \{\pi_k\}_{k\in\mathbb{N}}$ *is a family of structures, such that each* π_k *is the composition of* $p(k)$ $q(k)$*-time bounded structures. Then* $\overline{\pi}$ *is a polynomial time-bounded family of structure, bounded by the polynomial* $c_{\mathsf{comp}}pq$.

Proof. Lemma 9 guarantees that π_k is $c_{\mathsf{comp}}(p(k)q(k))$-time bounded.

The compatibility of two structures is preserved when we compose these structures with a third one.

Lemma 10. *Suppose* π_1 *and* π_2 *are comparable structures, and* π_3 *is a structure that is protocol-compatible with each of* π_1 *and* π_2.
Then $\pi_1 \| \pi_3$ *and* $\pi_2 \| \pi_3$ *are comparable structures.*

Proof. Write $\pi_1 = (\mathcal{A}_1, EAct_1)$, $\pi_2 = (\mathcal{A}_2, EAct_2)$, and $\pi_3 = (\mathcal{A}_3, EAct_3)$. We show the two conditions in the definition of comparability:

1. $EI_1 \cup EI_3 - (EO_1 \cup EO_3) = EI_2 \cup EI_3 - (EO_2 \cup EO_3)$.
   Since π_1 and π_2 are comparable structures, we know that $EI_1 = EI_2$ and $EO_1 = EO_2$. Let $a \in EI_1 \cup EI_3 - (EO_1 \cup EO_3)$. There are two cases:
   (a) $a \in EI_1 - EO_3$. Then $a \in EI_2$, so $a \in EI_2 - EO_3$. Since $a \in EI_2$, we have $a \notin EO_2$. So $a \in EI_2 \cup EI_3 - (EO_2 \cup EO_3)$, as needed.
   (b) $a \in EI_3 - EO_1$. Then $a \notin EO_2$, so $a \in EI_3 - EO_2$. Since $a \in EI_3$, we have $a \notin EO_3$. So $a \in EI_2 \cup EI_3 - (EO_2 \cup EO_3)$, as needed.
   The converse direction is similar.
2. $EO_1 \cup EO_3 = EO_2 \cup EO_3$.
   Since $EO_1 = EO_2$, this is immediate.

Time bounds of structures evolve as those of task-PIOAs when sets of output actions are hidden.

Lemma 11. *There exists a constant c_{hide} such that the following holds. Suppose π is a p-time-bounded structure, and S is a p'-time recognizable subset of the output actions of π. Then $hide(\pi, S)$ is a $c_{\mathsf{hide}}(p + p')$-time-bounded structure.*

Lemma 12. *Suppose $\overline{\pi}$ is a polynomial-time-bounded structure, and $\overline{S}$ is a polynomial-time recognizable family of subset of the output actions of $\overline{\pi}$. Then $hide(\overline{\pi}, \overline{S})$ is a polynomial-time-bounded structure.*

The proofs of these result are similar to those appearing in [20, Lemma 4.3 and 4.33].

C Adversary for Composed Structures

The following lemma relates signatures of adversaries and is used in the proof of Theorem 3.

Lemma 13. *Suppose ϕ and ψ are comparable structures, Adv is an adversary for ϕ, Sim is an adversary for ψ, and $hide(\phi \| Adv, AAct_\phi) \leq^{\mathsf{strong}}_{\mathsf{neg,pt}} hide(\psi \| Sim, AAct_\psi)$. Then, $O_{Adv} - AAct_\phi = O_{Sim} - AAct_\psi$, $I_{Adv} - AAct_\phi = I_{Sim} - AAct_\psi$, and $Ext_{Adv} - AAct_\phi = Ext_{Sim} - AAct_\psi$.*

Proof. Follows from the fact that ϕ and ψ are comparable structures, and that $hide(\phi \| Adv, AAct_\phi)$ and $hide(\psi \| Sim, AAct_\psi)$ must be comparable task-PIOAs.

Next we show that an adversary for the composition of several structures is an adversary of any of theses structures.

Lemma 14. *Suppose π and ϕ are compatible structures, and Adv is an adversary for $\pi \| \phi$. Then Adv is an adversary for ϕ. Also, if π and P are compatible structure families, and Adv is an adversary family for $\pi \| P$. Then Adv is an adversary family for P.*

Proof. Suppose π and ϕ are compatible structures, and Adv is an adversary for $\pi \| \phi$. We observe that the three conditions of Definition 4 are satisfied.

1. *Adv is compatible with ϕ.* This follows from the fact that Adv is compatible with $\pi \| \phi$.
2. $Ext_{Adv} \cap Ext_\phi \subseteq AAct_\phi$. Since Adv is an adversary for $\pi \| \phi$, we know that $Ext_{Adv} \cap (Ext_\pi \cup Ext_\phi) \subseteq AAct_\pi \cup AAct_\phi$. This implies that $Ext_{Adv} \cap Ext_\phi \subseteq AAct_\pi \cup AAct_\phi$. We observe now that $AAct_\pi \cap AAct_\phi = \emptyset$ and $AAct_\pi \cap EAct_\phi = \emptyset$, since π and ϕ are compatible structures. This implies that $AAct_\pi \cap Ext_\phi = \emptyset$, which in turn guarantees that $Ext_{Adv} \cap Ext_\phi \subseteq AAct_\phi$.

3. $AI_\phi \subseteq O_{Adv}$. Since Adv is an adversary for $\pi \| \phi$, we know that $(AAct_\phi \cup AAct_\pi) \cap ((I_\phi \cup I_\pi) - (O_\phi \cup O_\pi)) \subseteq O_{Adv}$. This first implies that $AAct_\phi \cap (I_\phi - (O_\phi \cup O_\pi)) \subseteq O_{Adv}$. Next, since $I_\phi \cap O_\phi = \emptyset$, we have that $AAct_\phi \cap (I_\phi - O_\pi)) \subseteq O_{Adv}$. By distributivity, we also have that $AAct_\phi \cap I_\phi - AAct_\phi \cap O_\pi \subseteq O_{Adv}$. The compatibility conditions of π and ϕ now imply that $AAct_\phi \cap O_\pi = \emptyset$, which provides the relation $AAct_\phi \cap I_\phi \subseteq O_{Adv}$, as needed.

The extension to structure families and adversary families is straightforward.

D Proof for Applications of the Composition Theorem

Proof (of Corollary 2). Let us write ϕ^i and ψ^i for $g^i(\phi)$ and $g^i(\psi)$ respectively. Since the g_i functions are just renaming functions, we have $\phi^i \leq_{\mathsf{SE}} \psi^i$ for every i.

Consider now, for every index i, the adversary family $Adv(\phi^i, f^i)$ for ϕ^i (following the definition used in Lemma 4), where the renamings f^i are such that $\phi^i \| Adv(\phi^i, f^i)$ is bounded by $r_1(i) \cdot s$ where r_1 and s are polynomials. The secure emulation relations above imply that there exist polynomial-time-bounded adversary families $Sim^1, Sim^2, \ldots$ for $\psi^1, \psi^2, \ldots$ (respectively) such that:

(a) for every i, the task-PIOA $\psi^i \| Sim^i$ is bounded by $r_2(i) \cdot s$, where r_2 is a polynomial (this can be stated because the renaming functions do not increase the length of action names too much, and because all Sim^i automata can be chosen identical up to action renaming), and

(b) $\forall q_1\ \exists q_2\ \forall p, q\ \exists \epsilon\ \forall i$
$\mathsf{hide}(\phi^i \| Adv(\phi^i, f^i), AAct_{\phi^i}) \leq_{q_1,q_2,p,q,\epsilon} \mathsf{hide}(\psi^i \| Sim^i, AAct_{\psi^i})$,
where q_1, q_2, p, q are polynomials, and ϵ is a negligible function.

As a result, by defining r as a non-decreasing polynomial majoring r_1 and r_2, we can apply Theorem 3 and obtain that $\widehat{\phi} \leq_{\mathsf{SE}} \widehat{\psi}$, as needed. □

Proof (of Corollary 3). Suppose $f^1, \ldots, f^B$ are renaming functions for the adversary actions of $\phi^1, \ldots, \phi^B$, such that the increase of the length of the action names of ϕ^i_k through f^i is bounded by some polynomial in k. Suppose further that $Adv(\phi^1, f^1), \ldots, Adv(\phi^B, f^B)$ are dummy adversary families as defined in Lemma 4.

Since B is constant, and since $\phi^i \leq_{\mathsf{SE}} \psi^i$ for every $i \in [B]$, there are adversary families $Sim^1, \ldots, Sim^B$ for $\psi^1, \ldots, \psi^B$ such that, $\forall q_1\ \exists q_2\ \forall p, q\ \exists \epsilon\ \forall i$ $\mathsf{hide}(\phi^i \| Adv(\phi^i, f^i), AAct_{\phi^i}) \leq_{q_1,q_2,p,q,\epsilon} \mathsf{hide}(\psi^i \| Sim^i, AAct_{\psi^i})$, where q_1, q_2, p, q are polynomials, and ϵ is a negligible function.

Now, the result follows of the use of Theorem 3 where r is a constant, s is a polynomial bounding the description of $\phi^i \| Adv(\phi^i, f^i)$ and $\psi^i \| Sim^i$ for every $i \in [B]$, and b is the constant B. □