Categorical composable cryptography

Anne Broadbent [1]   Martti Karvonen[2]

[1]University of Ottawa

[2]University College London

Workshop on Process Theory for Security Protocols and Cryptography
March 18 2024

Plan

- motivation: standard cryptography is not composable. Existing approaches to make it composable are a bit hacky/tedious/very complicated and seem to beg a categorical formalization

- main idea: cryptography as a resource theory — the resources are various functionalities (e.g. keys, channels etc) and transformations are given by protocols that build the target resource *securely* from the starting resources.

  - categories of correct resource conversions as a Grothendieck construction

  - correct and *secure* conversions as a subcategory

- example(ish): one-time-pad (OTP) as a transformation
  $OTP\colon key \otimes insecure\ channel \to secure\ channel$
  Security & correctness of OTP boil down to axioms of a Hopf algebra with an integral.

Real-world ideal-world paradigm

AKA simulation paradigm. Standard meta-approach for composable security.

Usual definition: a real protocol P securely realizes the ideal functionality F from the resource R if for any attack A on $P \circ R$ there is a simulator S on F such that $(A, P) \circ R$ is indistuingishable from $S \circ F$ by any (efficient) environment.

"Any bad thing that could happen during the protocol could also happen in the ideal world."

Usual ways of making this precise:

- Fixing a concrete low-level formalism for interactive computation (e.g. UC-security)
- Abstract cryptography and constructive cryptography — close to our work in spirit but technically different

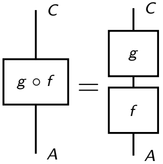
N+1th approach

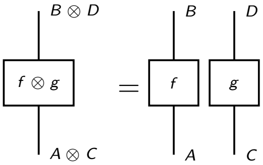
In our work we formalize the simulation paradigm over an arbitrary category (and a model of attacks). The main result is that protocols secure against a fixed attack model can be composed sequentially and in parallel. Some benefits:

- simulation-based security definitions are inherently composable, whether the model of computation is synchronous or not, classical or quantum etc.
- abstract attack models pave way for other kinds of attackers than malicious ones
- different notions of security (computational, finite-key regimen etc) fit in
- CT and the tools and connections it brings: in particular, string diagrams

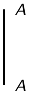
Recap on pictures

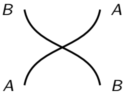
Let **C** be a symmetric monoidal category — concretely, you can think of (finite) sets and stochastic maps. We will depict a morphism as $\boxed{f}$, and composition and monoidal product as





Special morphisms get nicer pictures: identities and symmetries are





Resource theories

Roughly: An SMC where you mostly care whether a hom-set is empty or not.
Examples:

- Can these noisy channels be used to simulate a (almost) noiseless channel?
- Can this distribution be transformed to that one *deterministically*?
- Is there a LOCC-protocol that transforms this quantum state to that one?
- Any preordered commutative monoid.

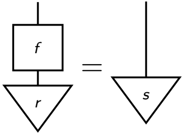
Many resource theories arise by taking the Grothendieck construction of $\mathbf{D} \xrightarrow{F} \mathbf{C} \xrightarrow{R} \mathbf{Set}$ where F interprets "free operations" in $\mathbf{C}$ and R gives for each $A \in \mathbf{C}$ the set $R(A)$ of resources of type $\mathbf{C}$.
Whenever RF is lax symmetric monoidal, $\int RF$ is a symmetric monoidal category, see '*Monoidal Grothendieck construction*'

Moeller & Vasilakopoulou, TAC 2020.

Running example: *n*-partite states

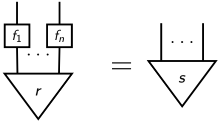
Resource theory of states: apply $\int$ to $\mathbf{C}_F \hookrightarrow \mathbf{C} \xrightarrow{\hom(I,-)} \mathbf{Set}$, where $\mathbf{C}_F \hookrightarrow \mathbf{C}$ is the subcategory of "free operations"

Objects are states of $\mathbf{C}$, and maps $r \to s$ are maps f in $\mathbf{C}_F$ such that



.

n-partite version: apply $\int$ to $\mathbf{C}_F^n \xrightarrow{\otimes} \mathbf{C} \to \mathbf{Set}$. Objects are of the form $((A_i)_{i=1}^n, r\colon I \to \bigotimes A_i)$. A map $(((A_i)_{i=1}^n, r) \to (((B_i)_{i=1}^n, s)$ is then a tuple $(f_i)_{i=1}^n$ that transforms r to s:

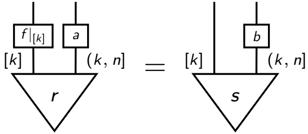


We think of this as a resource theory with n-parties who try to agree on actions $f_1, \dots f_n$ to transform some resource to another one.

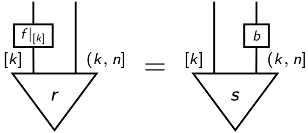
Security in the running example

Such a protocol is not necessarily secure—what if some subset of the parties does something else instead?

Assume the first k parties are honest and the last $n-k$ parties are dishonest. Then $(f_1, \dots f_n)$ is secure if for any a there is a b such that



It suffices to check this for the initial attack $\bigotimes_{k+1}^{n} \mathrm{id}$:



Security in the abstract

Usually a resource theory talks only about correct transformations

To add in security:

- need an attack model $\mathcal{A}$ that gives for each protocol f a collection $\mathcal{A}(f)$ of attacks on it, satisfying some axioms.
- security against $\mathcal{A}$: for each attack on the protocol there is an attack on the target with similar end-results

Definition

An attack model on **C** gives for each f a collection $\mathcal{A}(f)$ of morphisms in **C** such that (i) $\mathcal{A}(gf) = \mathcal{A}(g)\mathcal{A}(f)$ and (ii) $\mathcal{A}(g \otimes f) = \mathcal{A}(\mathrm{id}) \circ (\mathcal{A}(g) \otimes \mathcal{A}(f))$

E.g. malicious adversaries. honest-but-curious adversaries.

Note: If $f : A \to B$, we don't require $\mathcal{A}(f) \subset \mathbf{C}(A, B)$ — attackers don't care about our type system!

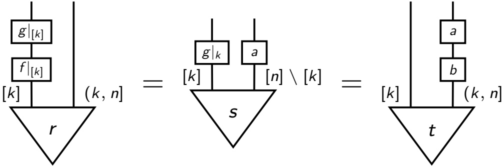
Composability

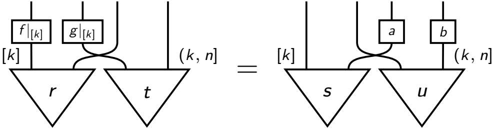
Theorem

Protocols that are secure against an attack model $\mathcal{A}$ are closed under composition (both $\circ$ and $\otimes$).

Proof.

$\circ$ and $\otimes$ are inherited form the ambient category—one just needs to check that they work. Here's the key steps for $\circ$ and $\otimes$ in the n-partite case with the first k parties honest





□

Security against multiple attack models

Corollary

Protocols secure against $\mathcal{A}_1, \ldots \mathcal{A}_k$ form a symmetric monoidal category

Proof.

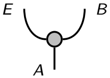
Symmetric monoidal subcategories are closed under intersection □

Example

Fix a family of subsets of n parties: protocols secure against each of these subsets behaving maliciously form an SMC. For instance, in MPC one often studies protocols secure against at most $n/2$ or $n/3$ malicious participants.

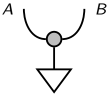
OTP: starting resources

Channel from Alice to Bob that leaks everything to Eve:

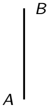


(Note: if instead the message goes via Eve (who may tamper with it), the analysis is different)

Shared random key:



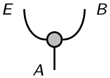
Target resource: a channel



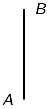
Free building blocks: local (efficient) computation

Insecure protocol

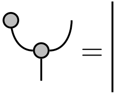
One way of transforming



to



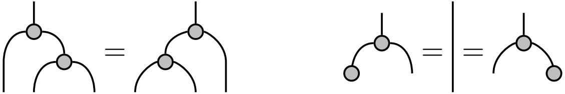
is by having Eve delete everything she receives, as



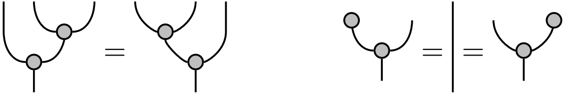
But this is not secure against Eve! No guarantees if Eve disobeys the protocol.

Local ingredients for OTP

A group structure on the message space: a multiplication with unit satisfying the following equations.

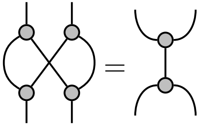


Note that copying and deleting satisfy similar equations

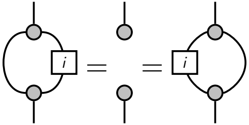


Rest of the group structure

In addition, multiplication and copying interact:

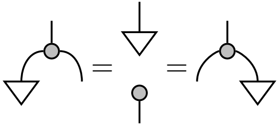


and the map $\boxed{i}$ giving inverses satisfies



Uniform randomness

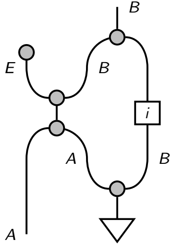
The key being uniformly random is captured by



"Adding uniform noise to a channel gives uniform noise"

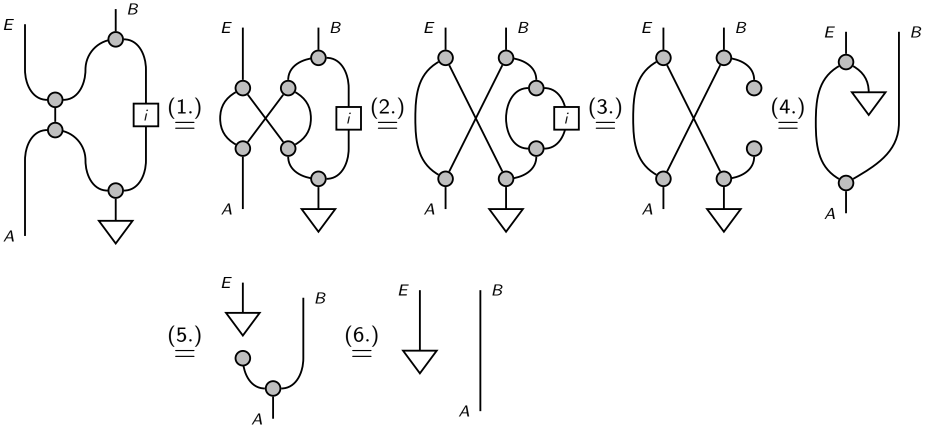
For the experts: a Hopf algebra with an integral in a symmetric monoidal category.

The protocol



Alice adds the key to her message, broadcasts it to Eve and Bob. Eve deletes her part and Bob adds the inverse of the key to recover the message.

Security of OTP



1. Bialgebra. 2. Associativity. 3. Antipode 4. Units 5. Random noise 6. Units.

More on OTP

In other words, anything Eve might learn from the ciphertext she could already compute without it, so this protocol is indeed a secure transformation against Eve.

Reusing keys is not a secure map $\mathit{key} \to \mathit{key} \otimes \mathit{key}$. However, a computationally secure PRNG will give a computationally secure way of constructing a long shared key from a short one. Composing these two results in *the stream cipher*, which is secure automatically as a composite of secure protocols inside our framework.

(Reasoning about computational security amounts to replacing equations with $\approx$ or working with a (pseudo)metric).

What next?

- just do composable cryptography within the framework
- rigorous comparisons with other composable frameworks
- string diagrams for “ordinary” (i.e., game-based) cryptographic reasoning?

Summary

We have a categorical framework where

- composability is guaranteed (also for computational security)
- attack models are general enough to cover various kinds of adversarial behavior (e.g. colluding vs independent attackers)
- string diagrams can be used to make existing (or new) pictures into rigorous proofs

Questions...

?

Broadbent A., MK, “Categorical composable cryptography”, FoSSaCS (2022),
arXiv:2105.05949
Broadbent A., MK, “Categorical composable cryptography: extended version”, LMCS (2023),
arXiv:2208.13232