Sheaf semantics of termination-insensitive noninterference

Jonathan Sterling (j.w.w. Robert Harper)

Aarhus University

International Conference on Formal Structures for Computation and Deduction (FSCD), 2022

open and closed modalities for SECURITY

Computer programs frequently must deal with **sensitive data**.

- your passwords
- your browser history
- your bank account number

The goal of **information flow control** is to provide tools to control the leakage of such data through program outputs.

Measured by **noninterference**: what kind of outputs can depend on what kinds of inputs?

State of the art: relational semantics that impose noninterference on an prior insecure model of computation (*e.g.* domains, op.sem.).

Our contribution: a new **intrinsic** denotational semantics for information flow control based on open and closed modalities from topos theory.

Noninterference via closed modalities

presheaves on a security (semi)lattice

We start with a meet semilattice $\mathbb{P}$ to be thought of as the collection of **security clearances**. For instance, we could have:

$$\mathbb{P} = \{\mathsf{low} \leq \mathsf{med} \leq \mathsf{high} \leq \top\}$$

The *presheaf topos* $\mathbf{P} = \hat{\mathbb{P}}$ is an ideal setting to develop a synthetic theory of security and redaction.

1. An open $\phi \in \mathcal{O}_{\mathbf{P}} = [\mathbb{P}^{\mathrm{op}}, \Omega]$ is a formal join of security clearances, which we may refer to as a **security policy**.

2. A (pre)sheaf $A \in \mathcal{S}_{\mathbf{P}} = [\mathbb{P}^{\mathrm{op}}, \mathbf{Set}]$ then has extent over each security policy $\phi \in \mathcal{O}_{\mathbf{P}}$, *i.e.* we may restrict A to $A_{|\phi} = \phi \times A \in \mathcal{S}_{\mathbf{P}}{\downarrow}\phi$.

   $A_{|\varphi}$ is the part of A that is visible under the policy φ.

   In particular, $A_{|\mathsf{med}} \in \mathcal{S}_{\mathbf{P}}{\downarrow}\mathsf{med}$ is the part of A that is visible to medium- and low-security observers.

open and closed subtopos

For each security policy $\phi \in \mathcal{O}_{\mathbf{P}}$, we have a pair of complementary open and closed subtopoi:

1. The **open** subtopos $\mathbf{U}_\phi \subseteq \mathbf{P}$ is defined by setting $\mathcal{S}_{\mathbf{U}_\phi}$ to be the full subcategory of $\mathcal{S}_{\mathbf{P}}$ spanned by sheaves A such that $A \longrightarrow A^\phi$ is an isomorphism, *i.e.* the slice $\mathcal{S}_{\mathbf{P}}\downarrow\phi$.

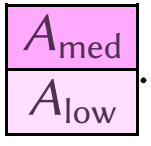
2. The **closed** subtopos $\mathbf{K}_\phi \subseteq \mathbf{P}$ is defined by setting $\mathcal{S}_{\mathbf{K}_\phi}$ to be the full subcategory of $\mathcal{S}_{\mathbf{P}}$ spaned by sheaves A such that $A \times \phi \longrightarrow \phi$ is an isomorphism.

We will visualize the modal operators corresponding to these subtopoi and their application to security.

a topo-logical viewpoint on redaction

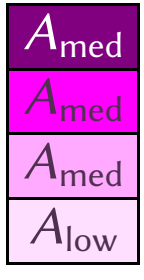
We may visualize an object $A \in \mathcal{S}_{\mathbf{P}}$ as a "pill" 

with components corresponding to what is visible at each security level.

Restricting into $\mathcal{S}_{\mathbf{P}}\downarrow\text{med}$ we have a truncated "pill": 

.

Restriction was the *inverse image* of the open immersion $\mathbf{P}_{\text{med}} \hookrightarrow \mathbf{P}$. The *direct image* extends our truncated "pill" as follows:



Thus all high-security data is redacted and replaced with medium security data.

the open modality as restriction

The round-trip $\mathcal{S}_{\mathbf{P}} \longrightarrow (\mathcal{S}_{\mathbf{P}}{\downarrow}\mathsf{med}) = \mathcal{S}_{\mathbf{U}_{\mathsf{med}}} \hookrightarrow \mathcal{S}_{\mathbf{P}}$ is the **open modality** for the (representable) proposition $\mathsf{med} \in \mathcal{O}_{\mathbf{P}}$ depicted below:

$$\mathsf{med} = \begin{array}{|c|}\hline \emptyset \\ \hline \emptyset \\ \hline \{*\} \\ \hline \{*\} \\ \hline \end{array}$$

Indeed, the following is not too hard to compute:

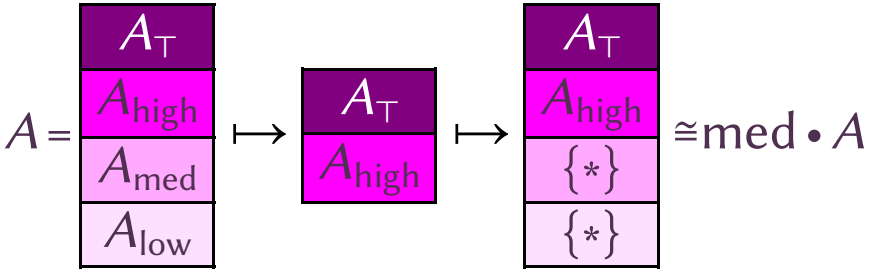
$$(\mathsf{med} \Longrightarrow A) \cong \begin{array}{|c|}\hline A_{\mathsf{med}} \\ \hline A_{\mathsf{med}} \\ \hline A_{\mathsf{med}} \\ \hline A_{\mathsf{low}} \\ \hline \end{array}$$

The open modality thus purges higher-security data.

sealing and closed subspace

The **closed complement** $\mathbf{K}_{\mathrm{med}}$ to $\mathbf{U}_{\mathrm{med}}$ is what parameterizes "high security (pre)sheaves", *i.e.* objects that only have extent visible to high and top security observers.

The closure operator corresponding to $\mathbf{K}_{\mathrm{med}} \subseteq \mathbf{P}$ causes a sheaf $A \in \mathcal{S}_{\mathbf{P}}$ to appear to contract to a point as far as medium and low security observers are concerned:



This is indeed the closed modality $\mathrm{med} \bullet A = \mathrm{med} \sqcup_{\mathrm{med} \times A} A$.

interaction between open and closed modalities

Here's the application to security. Suppose we want a type of integers that cannot leak to medium- and low-security clients.

1. We **seal** the type $\mathbb{Z}$ under the closed modality for $\mathsf{med} \in \mathcal{O}_{\mathbf{P}}$, *i.e.* we will use $\mathsf{med} \bullet \mathbb{Z}$.

2. Using the universal property of the closed modality, we can always unseal $\mathsf{med} \bullet \mathbb{Z}$ for consumption by higher security clients, *i.e.* types C that are contractible within med, *i.e.* med-connected or $(\mathsf{med} \bullet -)$-modal types:

$$(\mathsf{med} \bullet \mathbb{Z} \longrightarrow C) \cong (\mathbb{Z} \longrightarrow C)$$

Theorem 1. *Let* $f : \mathrm{med} \bullet A \longrightarrow 2$ *be any function; then* f *is* ***constant****.*

Proof. The type 2 is *constant*, *i.e.* we have $2 = \begin{array}{|c|}\hline \{0,1\} \\ \hline \{0,1\} \\ \hline \{0,1\} \\ \hline \{0,1\} \\ \hline \end{array}$.

Thus 2 is in particular $(\mathrm{med} \Longrightarrow -)$-modal and so we have:

$$\begin{aligned} (\mathrm{med} \bullet A \longrightarrow 2) &\cong (\mathrm{med} \bullet A \longrightarrow \mathrm{med} \Longrightarrow 2) \\ &\cong \mathrm{med} \Longrightarrow (\mathrm{med} \bullet A \longrightarrow 2) \\ &\cong \mathrm{med} \Longrightarrow 2 \\ &\cong 2 \end{aligned}$$

So we are done. □

Partial noninterference via synthetic domain theory

partiality via dominances

In the field of **programming languages** we are mainly interested in languages that support partial functions and general recursion.

In semantics, partiality means that we have a **dominance** $\top : 1 \rightarrowtail \Sigma$, *i.e.* a classifier for a collection of monomorphisms that will serve as the supports of partial functions.

The **partial map classifier** is the partial product of $\top : 1 \rightarrowtail \Sigma$, *i.e.*

$$A_\perp = \sum_{\phi : \Sigma} \phi \Longrightarrow A$$

A partial function $C \longrightarrow A_\perp$ is determined by a total function $U \longrightarrow A$ defined on a Σ-subobject $U \subseteq C$.

total and partial noninterference

With partiality, we can consider two kinds of noninterference:

1. **Total noninterference** means that any function $f : \mathsf{med} \bullet A \longrightarrow 2_\perp$ is constant. (*a.k.a.* "termination-sensitive noninterference")
2. **Partial noninterference** means that any function $f : \mathsf{med} \bullet A \longrightarrow 2$ is constant. (*a.k.a.* "termination-**in**sensitive noninterference")

From partial noninterference we conclude:

$$\forall x, y : \mathsf{med} \bullet A. \quad fx{\downarrow} \wedge fy{\downarrow} \longrightarrow fx = fy$$

Partial noninterference has been difficult to model satisfactorily in the past (*cf.* Abadi *et al.*). But follows immediately from standard **domain theoretic semantics** if we take **P** as our base topos.

synthetic domain theory: domain theory in a topos

Synthetic domain theory (Hyland) is to classical domain theory as synthetic differential geometry is to the theory of manifolds. Take a topos $\mathbf{D}$ equipped with a dominance $\top : 1 \rightarrowtail \Sigma$; the dominance is a **dualizing object** that associates to each object X a topology Σ^X.

We define a **predomain** to be an object X of $\mathcal{S}_{\mathbf{D}}$ satisfying a certain orthogonality condition; many possible conditions, but the weakest one that can possibly work is **well-completeness** (Longley):

Let $\omega_\perp \longrightarrow \omega$ be the initial algebra for the endofunctor $X \mapsto X_\perp$ and let $\bar{\omega} \longrightarrow \bar{\omega}_\perp$ be the final coalgebra; let $\omega \rightarrowtail \bar{\omega}$ be the canonical inclusion.

Definition 2. (Fiore, Plotkin) *An object X is a **predomain** when it is internally orthogonal to any pullback of $\omega \rightarrowtail \bar{\omega}$ along a Σ-monomorphism $U \rightarrowtail \bar{\omega}$.*

Orthogonality condition = closure under suprema of ω-chains.

synthetic domain theory over a base topos

Two classes of model of SDT have emerged: realizability models and sheaf models. We will focus on sheaf models.

The recipe of Fiore and Plotkin is to take sheaves on a small category of ordinary domains (*e.g.* ω-cpos, *&c.*). Fiore and Plotkin construct Grothendieck topos models of SDT over **Set**, but this recipe works over any base topos.

We construct a model of SDT over $\mathcal{S}_{\mathbf{P}}$, *i.e.* a geometric morphism $\delta : \mathbf{D} \longrightarrow \mathbf{P}$ such that $\Sigma = \delta^* \Omega_{\mathbf{P}}$.

The resulting notion of partial map includes not only convergence and divergence, but also divergence *above a certain security level*. There are computations that appear to converge for low-security clients, but can be seen to diverge by high-security clients.

declassification in the partial map classifier

Let A be modal for the closed modality $(\mathrm{med} \bullet -)$, *i.e.* suppose we have an algebra $\alpha : \mathrm{med} \bullet A \longrightarrow A$. Then we may define a function

$$\mathrm{dcls} : \mathrm{med} \bullet A_\perp \longrightarrow A_\perp$$

that "unseals" or "declassifies" a partial computation of type A.

$$\begin{array}{ccc} \mathrm{med} \bullet A & \xrightarrow{\alpha} & A \\ {\scriptstyle \mathrm{med} \bullet \eta}\downarrow & (\mathrm{cart.}) & \downarrow{\scriptstyle \eta} \\ \mathrm{med} \bullet A_\perp & \xrightarrow[\mathrm{dcls}]{} & A_\perp \end{array}$$

The subobject $\mathrm{med} \bullet A \rightarrowtail \mathrm{med} \bullet A_\perp$ lies in Σ because Σ is closed under the closed modality $(\mathrm{med} \bullet -)$: a form of **parallel computation**!

computational adequacy via synthetic Tait computability

We have defined a denotational semantics in synthetic domain theory for a programming language with modalities for security.

What is the relationship between equality of programs (the equational theory) and equality of their denotations?

Plotkin's **computational adequacy** lemma implies that denotational equivalence is a subrelation of observational equivalence.

We prove such a lemma for our language using **synthetic Tait computability**, an axiomatization of the internal language of Artin gluings.

gluing for computational adequacy, synthetically

Extend the internal language of a topos with:

1. disjoint propositions $\mathbf{\Phi}_{\text{syn}} \wedge \mathbf{\Phi}_{\text{sem}} = \bot$ and define $\mathbf{\Phi} = \mathbf{\Phi}_{\text{syn}} \vee \mathbf{\Phi}_{\text{sem}}$;
2. a **generic** model $[\![-]\!]_{\text{syn}}$ of $\mathbb{T}$ in the slice over $\mathbf{\Phi}_{\text{syn}}$;
3. the axioms of $\mathbb{P}$-indexed SDT in the slice over $\mathbf{\Phi}_{\text{sem}}$;
4. such that for each $l \in \mathbb{P}$ we have $[\![l]\!]_{\text{sem}} \leq \mathbf{\Phi} \bullet [\![l]\!]_{\text{syn}}$.

Under these axioms, we may construct a **synthetic logical relation** between the generic model and the denotational model that establishes computational adequacy.

Theorem 3. *If* $[\![\vdash e : \text{bool}_\bot]\!] = \eta b$ *then there exists* $\vdash v : \text{bool}$ *such that* $\vdash e \equiv \text{ret}(v) : \text{bool}$ *and* $[\![\vdash v : \text{bool}]\!] = b$.

Corollary 4. *Partial noninterference holds in the equational theory* $\mathbb{T}$.

summary

we have contributed an **intrinsic** denotational semantics of termination-insensitive noninterference: a counterpart to the usual **extrinsic**/relational accounts of noninterference.

abstract and synthetic methods used throughout to obtain simple and intuitive proofs of main results: **semantic noninterference**, **comptuational adequacy**, and **syntactic noninterference**.