Deterministic Channel Design for Minimum Leakage

Arthur Américo, MHR. Khouzani, Pasquale Malacaria

School of Electronic Engineering and Computer Science
Queen Mary University of London

32nd IEEE CSF – 28 June 2019

Introduction

Introduction

- Systems often need to leak some sensitive information to function correctly/efficiently

Introduction

- Systems often need to leak some sensitive information to function correctly/efficiently
  - A password checker always leaks information

Introduction

- Systems often need to leak some sensitive information to function correctly/efficiently
  - A password checker always leaks information
  - Eliminating all leakage from timing channels may lead to a substantial decrease in performance

Introduction

- Systems often need to leak some sensitive information to function correctly/efficiently
  - A password checker always leaks information
  - Eliminating all leakage from timing channels may lead to a substantial decrease in performance
- **Problem:** Find the system that minimizes information leakage while retaining functionality

Introduction

- Systems often need to leak some sensitive information to function correctly/efficiently
  - A password checker always leaks information
  - Eliminating all leakage from timing channels may lead to a substantial decrease in performance
- **Problem:** Find the system that minimizes information leakage while retaining functionality
- A general framework for this task was proposed in the CSF 2017 paper *Leakage Minimal Design: Universality, Limitations, and Applications*, by MHR. Khouzani and P. Malacaria

Introduction

- Systems often need to leak some sensitive information to function correctly/efficiently
  - A password checker always leaks information
  - Eliminating all leakage from timing channels may lead to a substantial decrease in performance
- **Problem:** Find the system that minimizes information leakage while retaining functionality
- A general framework for this task was proposed in the CSF 2017 paper *Leakage Minimal Design: Universality, Limitations, and Applications*, by MHR. Khouzani and P. Malacaria

Objective of this work

Study the application of this framework to deterministic systems

Preliminaries

Quantitative Information Flow

- A secret value is taken from a set $\mathcal{X} = \{x_1, \ldots, x_n\}$ according to a distribution π

Quantitative Information Flow

- A secret value is taken from a set $\mathcal{X} = \{x_1, \ldots, x_n\}$ according to a distribution π
- A system takes the secret value as input and produces an observable behaviour (or simply observable) in $\mathcal{Y} = \{y_1, \ldots, y_m\}$

Quantitative Information Flow

- A secret value is taken from a set $\mathcal{X} = \{x_1, \ldots, x_n\}$ according to a distribution π
- A system takes the secret value as input and produces an observable behaviour (or simply observable) in $\mathcal{Y} = \{y_1, \ldots, y_m\}$
- An adversary, observing the behaviour of the system, may obtain some information about the secret value

Systems as Channels

- A system with inputs in $\mathcal{X}$ and observables in $\mathcal{Y}$ is modelled by a channel $C : \mathcal{X} \to \mathcal{Y}$.

Systems as Channels

- A system with inputs in $\mathcal{X}$ and observables in $\mathcal{Y}$ is modelled by a channel $C : \mathcal{X} \rightarrow \mathcal{Y}$.
- $C(x, y)$ is the conditional probability that $y \in \mathcal{Y}$ will be produced given that the secret value is $x \in \mathcal{X}$

Systems as Channels

- A system with inputs in $\mathcal{X}$ and observables in $\mathcal{Y}$ is modelled by a channel $C : \mathcal{X} \to \mathcal{Y}$.
- $C(x, y)$ is the conditional probability that $y \in \mathcal{Y}$ will be produced given that the secret value is $x \in \mathcal{X}$
  - $C(x, y) > 0$ $\qquad$ $\sum_y C(x, y) = 1$

C	y_1	y_2	y_3	y_4
x_1	$1/2$	$1/4$	$1/8$	$1/8$
x_2	$1/4$	$1/2$	$1/4$	0
x_3	1	0	0	0

Systems as Channels

- A system with inputs in $\mathcal{X}$ and observables in $\mathcal{Y}$ is modelled by a channel $C : \mathcal{X} \to \mathcal{Y}$.
- $C(x, y)$ is the conditional probability that $y \in \mathcal{Y}$ will be produced given that the secret value is $x \in \mathcal{X}$
  - $C(x, y) > 0$ $\qquad\qquad \sum_y C(x, y) = 1$

C	y_1	y_2	y_3	y_4
x_1	$1/2$	$1/4$	$1/8$	$1/8$
x_2	$1/4$	$1/2$	$1/4$	0
x_3	1	0	0	0

- In this work we focus on deterministic channels: $C(x, y) \in \{0, 1\}$

Systems as Channels

- A system with inputs in $\mathcal{X}$ and observables in $\mathcal{Y}$ is modelled by a channel $C : \mathcal{X} \to \mathcal{Y}$.
- $C(x, y)$ is the conditional probability that $y \in \mathcal{Y}$ will be produced given that the secret value is $x \in \mathcal{X}$
  - $C(x, y) > 0$ $\qquad \sum_y C(x, y) = 1$

C	y_1	y_2	y_3	y_4
x_1	0	1	0	0
x_2	0	1	0	0
x_3	1	0	0	0

- In this work we focus on deterministic channels: $C(x, y) \in \{0, 1\}$

How is information leaked?

- The adversary knows π and C

π
$1/3$
$1/4$
$1/4$
$1/6$

C	y_1	y_2
x_1	1	0
x_2	0	1
x_3	0	1
x_3	1	0

How is information leaked?

- The adversary knows π and C
- Joint distribution $p(x,y) = \pi(x)C(x,y)$

p	y_1	y_2
x_1	$1/3$	0
x_2	0	$1/4$
x_3	0	$1/4$
x_4	$1/6$	0

How is information leaked?

- The adversary knows π and C
- Joint distribution $p(x, y) = \pi(x)C(x, y)$
- Marginal distribution $p(y) = \sum_{x \in \mathcal{X}} p(x, y)$

p	y_1	y_2
x_1	$1/3$	0
x_2	0	$1/4$
x_3	0	$1/4$
x_4	$1/6$	0

$p(y_1) = 1/2$

$p(y_2) = 1/2$

How is information leaked?

- The adversary knows π and C
- Joint distribution $p(x, y) = \pi(x)C(x, y)$
- Marginal distribution $p(y) = \sum_{x \in \mathcal{X}} p(x, y)$
- Posterior distributions $p_{\mathcal{X}|y}(x) = \frac{p(x,y)}{p(y)}$

	$p_{\mathcal{X}\|y_1}$	$p_{\mathcal{X}\|y_2}$
x_1	$2/3$	0
x_2	0	$1/2$
x_3	0	$1/2$
x_4	$1/3$	0

$p(y_1) = 1/2$

$p(y_2) = 1/2$

How is information leaked?

- The adversary knows π and C
- Joint distribution $p(x,y) = \pi(x)C(x,y)$
- Marginal distribution $p(y) = \sum_{x \in \mathcal{X}} p(x,y)$
- Posterior distributions $p_{\mathcal{X}|y}(x) = \frac{p(x,y)}{p(y)}$

	$p_{\mathcal{X}\|y_1}$	$p_{\mathcal{X}\|y_2}$
x_1	$^2/_3$	0
x_2	0	$^1/_2$
x_3	0	$^1/_2$
x_4	$^1/_3$	0

$$p(y_1) = {}^1/_2$$

$$p(y_2) = {}^1/_2$$

- By observing y, the adversary updates the distribution from π to $p_{\mathcal{X}|y}$

Quantifying Information Leakage

- An entropy measure H reflects how uncertain an adversary is about the secret value

Quantifying Information Leakage

- An entropy measure H reflects how uncertain an adversary is about the secret value
- Many Choices: Shannon Entropy (H_1), min-entropy (H_∞), guessing entropy (H_G) ...

Quantifying Information Leakage

- An entropy measure H reflects how uncertain an adversary is about the secret value
- Many Choices: Shannon Entropy (H_1), min-entropy (H_∞), guessing entropy (H_G) ...
- $H(\pi)$ = initial uncertainty

Quantifying Information Leakage

- An entropy measure H reflects how uncertain an adversary is about the secret value
- Many Choices: Shannon Entropy (H_1), min-entropy (H_∞), guessing entropy (H_G) ...
- $H(\pi) =$ initial uncertainty
- $H(\pi, C) =$ uncertainty after execution

Quantifying Information Leakage

- An entropy measure H reflects how uncertain an adversary is about the secret value
- Many Choices: Shannon Entropy (H_1), min-entropy (H_∞), guessing entropy (H_G) ...
- $H(\pi)$ = initial uncertainty
- $H(\pi, C)$ = uncertainty after execution
- Leakage $= H(\pi) - H(\pi, C)$

Deterministic Channel Design

- Leakage $= H(\pi) - H(\pi, C)$

Deterministic Channel Design

- Leakage $= H(\pi) - H(\pi, C)$

Deterministic Channel Design Problem

Given π and a reasonable entropy measure H, find the deterministic channel C that maximizes $H(\pi, C)$, respecting some operational constraints

Deterministic Channel Design

- Leakage $= H(\pi) - H(\pi, C)$

Deterministic Channel Design Problem

Given π and a reasonable entropy measure H, find the deterministic channel C that maximizes $H(\pi, C)$, respecting some operational constraints

- Maximize $H(\pi, C) =$ Minimize Leakage

Deterministic Channel Design

- Leakage $= H(\pi) - H(\pi, C)$

Deterministic Channel Design Problem

Given π and a reasonable entropy measure H, find the deterministic channel C that maximizes $H(\pi, C)$, respecting some operational constraints

- Maximize $H(\pi, C) =$ Minimize Leakage
- What is a reasonable entropy?

Deterministic Channel Design

- Leakage $= H(\pi) - H(\pi, C)$

Deterministic Channel Design Problem

Given π and a reasonable entropy measure H, find the deterministic channel C that maximizes $H(\pi, C)$, respecting some operational constraints

- Maximize $H(\pi, C)$ = Minimize Leakage
- What is a reasonable entropy?
- How should we model operational constraints?

What is a Reasonable Entropy?

- A entropy H is core-concave if there is η, F such that
  - $H(\pi) = \eta(F(\pi))$
  - F is a real valued, continuous and concave function
  - $\eta : I \to \mathbb{R}$ is continuous and increasing

What is a Reasonable Entropy?

- A entropy H is core-concave if there is η, F such that
  - $H(\pi) = \eta(F(\pi))$
  - F is a real valued, continuous and concave function
  - $\eta : I \to \mathbb{R}$ is continuous and increasing
- Prior entropy $H(\pi) = \eta(F(\pi))$

What is a Reasonable Entropy?

- A entropy H is core-concave if there is η, F such that
  - $H(\pi) = \eta(F(\pi))$
  - F is a real valued, continuous and concave function
  - $\eta : I \to \mathbb{R}$ is continuous and increasing
- Prior entropy $H(\pi) = \eta(F(\pi))$
- Posterior entropy

$$H(\pi, C) = \eta\left(\sum_y p(y) F(p_{\mathcal{X}|y})\right)$$

What is a Reasonable Entropy?

- A entropy H is core-concave if there is η, F such that
  - $H(\pi) = \eta(F(\pi))$
  - F is a real valued, continuous and concave function
  - $\eta : I \to \mathbb{R}$ is continuous and increasing
- Prior entropy $H(\pi) = \eta(F(\pi))$
- Posterior entropy

$$H(\pi, C) = \eta\left(\sum_{y} p(y) F(p_{\mathcal{X}|y})\right)$$

- Generalizes most entropy measures in QIF

How Should We Model Operational Constraints?

- Hard constraints: A set $\Omega \subset \mathcal{X} \times \mathcal{Y}$ of which observables can be produced for each secret.
  - $C(x, y) > 0 \implies (x, y) \in \Omega$

How Should We Model Operational Constraints?

- Hard constraints: A set $\Omega \subset \mathcal{X} \times \mathcal{Y}$ of which observables can be produced for each secret.
  - $C(x, y) > 0 \implies (x, y) \in \Omega$

$$\Omega = \{(x_1, y_1), (x_1, y_2), (x_2, y_1), (x_2, y_3), (x_3, y_3)\}$$

How Should We Model Operational Constraints?

- Hard constraints: A set $\Omega \subset \mathcal{X} \times \mathcal{Y}$ of which observables can be produced for each secret.
  - $C(x, y) > 0 \implies (x, y) \in \Omega$

$$\Omega = \{(x_1, y_1), (x_1, y_2), (x_2, y_1), (x_2, y_3), (x_3, y_3)\}$$

C	y_1	y_2	y_3
x_1	?	?	0
x_2	?	0	?
x_3	0	0	?

How Should We Model Operational Constraints?

- Hard constraints: A set $\Omega \subset \mathcal{X} \times \mathcal{Y}$ of which observables can be produced for each secret.
  - $C(x, y) > 0 \implies (x, y) \in \Omega$

$$\Omega = \{(x_1, y_1), (x_1, y_2), (x_2, y_1), (x_2, y_3), (x_3, y_3)\}$$

C	y_1	y_2	y_3
x_1	1	0	0
x_2	1	0	0
x_3	0	0	1

How Should We Model Operational Constraints?

- Hard constraints: A set $\Omega \subset \mathcal{X} \times \mathcal{Y}$ of which observables can be produced for each secret.
  - $C(x, y) > 0 \implies (x, y) \in \Omega$

$$\Omega = \{(x_1, y_1), (x_1, y_2), (x_2, y_1), (x_2, y_3), (x_3, y_3)\}$$

C	y_1	y_2	y_3
x_1	0	1	0
x_2	0	0	1
x_3	0	0	1

How to model operational constraints?

- Soft constraints: A function $u : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ gives the "utility" of each pair of secret and observable

How to model operational constraints?

- Soft constraints: A function $u : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ gives the "utility" of each pair of secret and observable
  - Execution time, difference between real and reported data, ...

How to model operational constraints?

- Soft constraints: A function $u : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ gives the "utility" of each pair of secret and observable
  - Execution time, difference between real and reported data, ...
- Constraint: $\mathbb{E}[u] = \sum_{x,y} \pi(x) C(x,y) u(x,y) \geq u_{min}$

The general framework for the Channel Design Problem

(Probabilistic) Channel Design Problem (Khouzani and Malacaria, CSF 2017)

Find channel $C : \mathcal{X} \to \mathcal{Y}$ that maximizes $H(\pi, C)$ subject to

- $C(x, y) > 0 \implies (x, y) \in \Omega$
- $\mathbb{E}[u] \geq u_{min}$

.

The general framework for the Channel Design Problem

(Probabilistic) Channel Design Problem (Khouzani and Malacaria, CSF 2017)

Find channel $C : \mathcal{X} \rightarrow \mathcal{Y}$ that maximizes $H(\pi, C)$ subject to

- $C(x, y) > 0 \implies (x, y) \in \Omega$
- $\mathbb{E}[u] \geq u_{min}$

.

- Solved by convex programming (Karush-Kuhn Tucker conditions)

The Deterministic Channel Design Problem

The Deterministic Channel Design Problem

Deterministic Channel Design Problem:

Find channel $C : \mathcal{X} \to \mathcal{Y}$ that maximizes $H(\pi, C)$ subject to

- $C(x, y) \in \{0, 1\}$
- $C(x, y) > 0 \implies (x, y) \in \Omega$
- $\mathbb{E}[u] \geq u_{min}$

.

NP-Hardness

Theorem

The Deterministic Channel Design Problem is NP-Hard

NP-Hardness

Theorem

The Deterministic Channel Design Problem is NP-Hard

Proof: reduction from the Set Covering Problem

NP-Hardness

Theorem

The Deterministic Channel Design Problem is NP-Hard

Proof: reduction from the Set Covering Problem

Let $\mathcal{U}$ be a finite set and $\mathcal{C} \subset 2^{\mathcal{U}}$ a collection of subsets of $\mathcal{U}$

NP-Hardness

Theorem

The Deterministic Channel Design Problem is NP-Hard

Proof: reduction from the Set Covering Problem

Let $\mathcal{U}$ be a finite set and $\mathcal{C} \subset 2^{\mathcal{U}}$ a collection of subsets of $\mathcal{U}$

There is a subcollection of $\mathcal{C}$ of size $k > 0$ that covers $\mathcal{U}$

NP-Hardness

Theorem

The Deterministic Channel Design Problem is NP-Hard

Proof: reduction from the Set Covering Problem

Let $\mathcal{U}$ be a finite set and $\mathcal{C} \subset 2^{\mathcal{U}}$ a collection of subsets of $\mathcal{U}$

There is a subcollection of $\mathcal{C}$ of size $k > 0$ that covers $\mathcal{U}$

$$\Updownarrow$$

NP-Hardness

Theorem

The Deterministic Channel Design Problem is NP-Hard

Proof: reduction from the Set Covering Problem

Let $\mathcal{U}$ be a finite set and $\mathcal{C} \subset 2^{\mathcal{U}}$ a collection of subsets of $\mathcal{U}$

There is a subcollection of $\mathcal{C}$ of size $k > 0$ that covers $\mathcal{U}$

$$\Updownarrow$$

There is a channel $C : \mathcal{U} \to \mathcal{C}$, with $H_\infty(\pi_u, C) \geq -\log \frac{k}{|\mathcal{U}|}$

NP-Hardness

Theorem

The Deterministic Channel Design Problem is NP-Hard

Proof: reduction from the Set Covering Problem

Let $\mathcal{U}$ be a finite set and $\mathcal{C} \subset 2^{\mathcal{U}}$ a collection of subsets of $\mathcal{U}$

There is a subcollection of $\mathcal{C}$ of size $k > 0$ that covers $\mathcal{U}$

$$\Updownarrow$$

There is a channel $C : \mathcal{U} \to \mathcal{C}$, with $H_\infty(\pi_u, C) \geq -\log \frac{k}{|\mathcal{U}|}$

(π_u is the uniform distribution, and $\Omega = \{(x, y) \mid x \in y\}$)

Universality of the Solution

- The choice of entropy measure depends on the adversary's interests and probabilities.

Universality of the Solution

- The choice of entropy measure depends on the adversary's interests and probabilities.
- This may be outside of the designer's control...

Universality of the Solution

- The choice of entropy measure depends on the adversary's interests and probabilities.
- This may be outside of the designer's control...
- Thus, a desirable property is universality: there is C that is a solution for all core-concave entropies

Universality of the Solution

- The choice of entropy measure depends on the adversary's interests and probabilities.
- This may be outside of the designer's control...
- Thus, a desirable property is universality: there is C that is a solution for all core-concave entropies

Theorem

In general, the Deterministic Channel Design Problem does not satisfy universality

Universality of the Solution

Theorem

In general, the Deterministic Channel Design Problem does not satisfy universality

Universality of the Solution

Theorem

In general, the Deterministic Channel Design Problem does not satisfy universality

Proof Let $\Omega = \{(x_1, y_1), (x_2, y_1), (x_1, y_2), (x_3, y_2), (x_2, y_3), (x_4, y_3)\}$

π
0.35
0.35
0.15
0.15

C	y_1	y_2	y_3
x_1	?	?	0
x_2	?	0	?
x_3	0	?	0
x_4	0	0	?

Universality of the Solution

Theorem

In general, the Deterministic Channel Design Problem does not satisfy universality

Proof Let $\Omega = \{(x_1, y_1), (x_2, y_1), (x_1, y_2), (x_3, y_2), (x_2, y_3), (x_4, y_3)\}$

π
0.35
0.35
0.15
0.15

C	y_1	y_2	y_3
x_1	1	0	0
x_2	1	0	0
x_3	0	1	0
x_4	0	0	1

Optimal for min-entropy

Universality of the Solution

Theorem

In general, the Deterministic Channel Design Problem does not satisfy universality

Proof Let $\Omega = \{(x_1, y_1), (x_2, y_1), (x_1, y_2), (x_3, y_2), (x_2, y_3), (x_4, y_3)\}$

π
0.35
0.35
0.15
0.15

C	y_1	y_2	y_3
x_1	0	1	0
x_2	0	0	1
x_3	0	1	0
x_4	0	0	1

Optimal for Shannon entropy

The complete k-hypergraph problem

The Complete k-hypergraph Problem:

- The Complete k-hypergraph Problem: at most k secret values can be mapped to each observable
  - $\mathcal{Y} = \{\mathcal{A} \subset \mathcal{X} \,|\, |\mathcal{A}| \leq k\}, \qquad \Omega = \{(x, y) \,|\, x \in y\}$.

The Complete k-hypergraph Problem:

- The Complete k-hypergraph Problem: at most k secret values can be mapped to each observable
  - $\mathcal{Y} = \{\mathcal{A} \subset \mathcal{X} \,|\, |\mathcal{A}| \leq k\}, \qquad \Omega = \{(x, y) \,|\, x \in y\}$.
- **Result:** There is a greedy solution to a subset of core-concave entropies, called leakage-supermodular
  - Includes most entropies used in QIF: Shannon entropy, min-entropy, guessing entropy...

Leakage-Supermodularity: Supermodular Functions

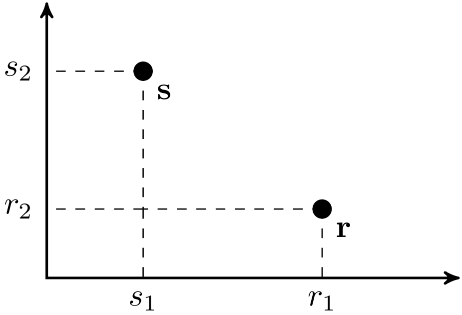
- Lattice on $\mathbb{R}^n_{\geq 0}$: Let $\mathbf{r} = (r_1, \ldots, r_n)$,$\mathbf{s} = (s_1, \ldots, s_n) \in \mathbb{R}^n_{\geq 0}$
  - Join: $\mathbf{r} \vee \mathbf{s} = (\max(r_1, s_1), \ldots, \max(r_n, s_n))$
  - Meet: $\mathbf{r} \wedge \mathbf{s} = (\min(r_1, s_1), \ldots, \min(r_n, s_n))$

Leakage-Supermodularity: Supermodular Functions

- Lattice on $\mathbb{R}^n_{\geq 0}$: Let $\mathbf{r} = (r_1, \ldots, r_n)$,$\mathbf{s} = (s_1, \ldots, s_n) \in \mathbb{R}^n_{\geq 0}$
  - Join: $\mathbf{r} \vee \mathbf{s} = (\max(r_1, s_1), \ldots, \max(r_n, s_n))$
  - Meet: $\mathbf{r} \wedge \mathbf{s} = (\min(r_1, s_1), \ldots, \min(r_n, s_n))$

Example ($n = 2$):

$$\begin{aligned} \mathbf{r} &= (r_1, r_2) \\ \mathbf{s} &= (s_1, s_2) \end{aligned}$$

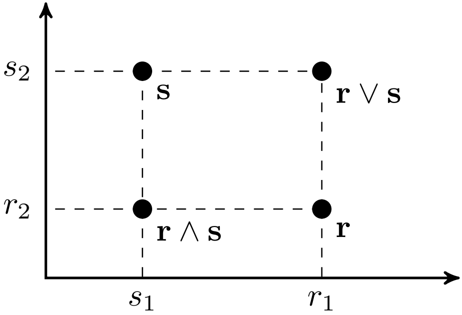


Leakage-Supermodularity: Supermodular Functions

- Lattice on $\mathbb{R}^n_{\geq 0}$: Let $\mathbf{r} = (r_1, \ldots, r_n)$,$\mathbf{s} = (s_1, \ldots, s_n) \in \mathbb{R}^n_{\geq 0}$
  - Join: $\mathbf{r} \vee \mathbf{s} = (\max(r_1, s_1), \ldots, \max(r_n, s_n))$
  - Meet: $\mathbf{r} \wedge \mathbf{s} = (\min(r_1, s_1), \ldots, \min(r_n, s_n))$

Example ($n = 2$):

$$\begin{aligned} \mathbf{r} &= (r_1, r_2) \\ \mathbf{s} &= (s_1, s_2) \end{aligned}$$



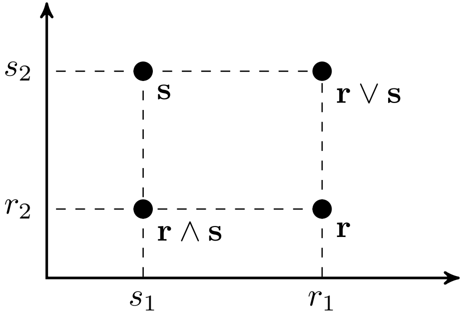
Leakage-Supermodularity: Supermodular Functions

- Lattice on $\mathbb{R}^n_{\geq 0}$: Let $\mathbf{r} = (r_1, \ldots, r_n)$,$\mathbf{s} = (s_1, \ldots, s_n) \in \mathbb{R}^n_{\geq 0}$
  - Join: $\mathbf{r} \vee \mathbf{s} = (\max(r_1, s_1), \ldots, \max(r_n, s_n))$
  - Meet: $\mathbf{r} \wedge \mathbf{s} = (\min(r_1, s_1), \ldots, \min(r_n, s_n))$
- A function $\phi : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$ is Supermodular if

$$\phi(\mathbf{r} \vee \mathbf{s}) + \phi(\mathbf{r} \wedge \mathbf{s}) \geq \phi(\mathbf{r}) + \phi(\mathbf{s})$$

Example ($n = 2$):

$$\mathbf{r} = (r_1, r_2)$$
$$\mathbf{s} = (s_1, s_2)$$



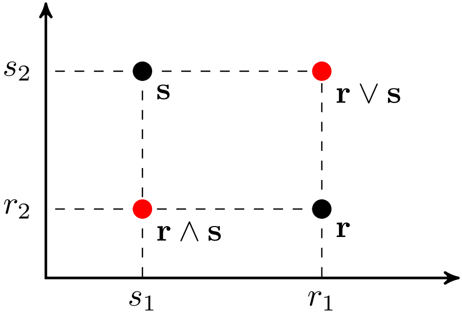
Leakage-Supermodularity: Supermodular Functions

- Lattice on $\mathbb{R}^n_{\geq 0}$: Let $\mathbf{r} = (r_1, \ldots, r_n)$,$\mathbf{s} = (s_1, \ldots, s_n) \in \mathbb{R}^n_{\geq 0}$
  - Join: $\mathbf{r} \vee \mathbf{s} = (\max(r_1, s_1), \ldots, \max(r_n, s_n))$
  - Meet: $\mathbf{r} \wedge \mathbf{s} = (\min(r_1, s_1), \ldots, \min(r_n, s_n))$
- A function $\phi : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$ is Supermodular if

$$\phi(\mathbf{r} \vee \mathbf{s}) + \phi(\mathbf{r} \wedge \mathbf{s}) \geq \phi(\mathbf{r}) + \phi(\mathbf{s})$$

Example ($n = 2$):

$$\mathbf{r} = (r_1, r_2)$$
$$\mathbf{s} = (s_1, s_2)$$



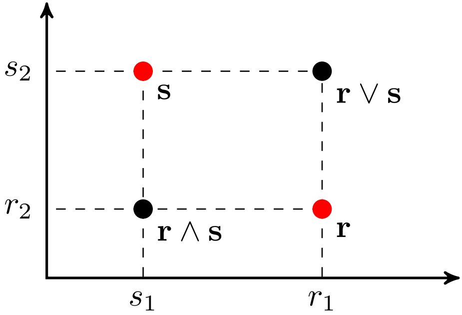
Leakage-Supermodularity: Supermodular Functions

- Lattice on $\mathbb{R}^n_{\geq 0}$: Let $\mathbf{r} = (r_1, \ldots, r_n)$,$\mathbf{s} = (s_1, \ldots, s_n) \in \mathbb{R}^n_{\geq 0}$
  - Join: $\mathbf{r} \vee \mathbf{s} = (\max(r_1, s_1), \ldots, \max(r_n, s_n))$
  - Meet: $\mathbf{r} \wedge \mathbf{s} = (\min(r_1, s_1), \ldots, \min(r_n, s_n))$
- A function $\phi : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$ is Supermodular if

$$\phi(\mathbf{r} \vee \mathbf{s}) + \phi(\mathbf{r} \wedge \mathbf{s}) \geq \phi(\mathbf{r}) + \phi(\mathbf{s})$$

Example ($n = 2$):

$$\mathbf{r} = (r_1, r_2)$$
$$\mathbf{s} = (s_1, s_2)$$



Leakage-Supermodular Entropies

- For now on, we restrict our attention to entropies that are
  - Symmetric: $H(\pi_1, \ldots, \pi_n) = H(\pi_{\phi(1)}, \ldots, \pi_{\phi(n)})$ for all permutations ϕ
  - Expansible: $H(\pi_1, \ldots, \pi_n, 0) = H(\pi_1, \ldots, \pi_n)$

Leakage-Supermodular Entropies

- For now on, we restrict our attention to entropies that are
  - Symmetric: $H(\pi_1, \ldots, \pi_n) = H(\pi_{\phi(1)}, \ldots, \pi_{\phi(n)})$ for all permutations ϕ
  - Expansible: $H(\pi_1, \ldots, \pi_n, 0) = H(\pi_1, \ldots, \pi_n)$
- Given a core-concave H for some η, F, define $G_F : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$

$$G_F(r_1, \ldots, r_n) = \left(\sum_i r_i\right) F\left(\frac{r_1}{\sum_i r_i}, \ldots, \frac{r_n}{\sum_i r_i}\right)$$

Leakage-Supermodular Entropies

- For now on, we restrict our attention to entropies that are
  - Symmetric: $H(\pi_1, \ldots, \pi_n) = H(\pi_{\phi(1)}, \ldots, \pi_{\phi(n)})$ for all permutations ϕ
  - Expansible: $H(\pi_1, \ldots, \pi_n, 0) = H(\pi_1, \ldots, \pi_n)$
- Given a core-concave H for some η, F, define $G_F : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$

$$G_F(r_1, \ldots, r_n) = \left(\sum_i r_i\right) F\left(\frac{r_1}{\sum_i r_i}, \ldots, \frac{r_n}{\sum_i r_i}\right)$$

- H is leakage-supermodular if G_F is supermodular

Leakage-Supermodular Entropies

- For now on, we restrict our attention to entropies that are
  - Symmetric: $H(\pi_1, \ldots, \pi_n) = H(\pi_{\phi(1)}, \ldots, \pi_{\phi(n)})$ for all permutations ϕ
  - Expansible: $H(\pi_1, \ldots, \pi_n, 0) = H(\pi_1, \ldots, \pi_n)$
- Given a core-concave H for some η, F, define $G_F : \mathbb{R}^n_{\geq 0} \to \mathbb{R}$

$$G_F(r_1, \ldots, r_n) = \left(\sum_i r_i\right) F\left(\frac{r_1}{\sum_i r_i}, \ldots, \frac{r_n}{\sum_i r_i}\right)$$

- H is leakage-supermodular if G_F is supermodular

Theorem

Shannon entropy, min-entropy, guessing entropy and Arimoto-Rényi entropies are leakage-supermodular

Leakage-Supermodular Entropies

- Relation to leakage:

$$H(\pi, C) = \eta \left(\sum_y G_F \left(p(x_1, y), \ldots, p(x_n, y) \right) \right)$$

Leakage-Supermodular Entropies

- Relation to leakage:

$$H(\pi, C) = \eta \left(\sum_y G_F \left(p(x_1, y), \ldots, p(x_n, y) \right) \right)$$

π
$1/3$
$1/4$
$1/4$
$1/6$

C	y_1	y_2
x_1	1	0
x_2	0	1
x_3	0	1
x_3	1	0

Leakage-Supermodular Entropies

- Relation to leakage:

$$H(\pi, C) = \eta \left(\sum_{y} G_F \left(p(x_1, y), \ldots, p(x_n, y) \right) \right)$$

p	y_1	y_2
x_1	$1/3$	0
x_2	0	$1/4$
x_3	0	$1/4$
x_4	$1/6$	0

Leakage-Supermodular Entropies

- Relation to leakage:

$$H(\pi, C) = \eta\left(\sum_y G_F\left(p(x_1, y), \ldots, p(x_n, y)\right)\right)$$

p	y_1	y_2
x_1	$1/3$	0
x_2	0	$1/4$
x_3	0	$1/4$
x_4	$1/6$	0

$$H(\pi, C) = \eta\Big(G_F(1/3, 0, 0, 1/6) + G_F(0, 1/4, 1/4, 0)\Big)$$

A greedy solution

The k-hypergraph Problem

Let $\mathcal{Y} = \{\mathcal{A} \subset \mathcal{X} \mid |\mathcal{A}| \leq k\}$. Find $C : \mathcal{X} \to \mathcal{Y}$ that maximizes $H(\pi, C)$, subject to $\Omega = \{(x, y) \in \mathcal{X} \times \mathcal{Y} \mid x \in y\}$

A greedy solution

The k-hypergraph Problem

Let $\mathcal{Y} = \{\mathcal{A} \subset \mathcal{X} \mid |\mathcal{A}| \leq k\}$. Find $C : \mathcal{X} \to \mathcal{Y}$ that maximizes $H(\pi, C)$, subject to $\Omega = \{(x, y) \in \mathcal{X} \times \mathcal{Y} \mid x \in y\}$

- Order $\mathcal{X} = \{x_1, \ldots, x_n\}$ such that $\pi(x_1) \geq \cdots \geq \pi(x_n)$

A greedy solution

The k-hypergraph Problem

Let $\mathcal{Y} = \{\mathcal{A} \subset \mathcal{X} \mid |\mathcal{A}| \leq k\}$. Find $C : \mathcal{X} \to \mathcal{Y}$ that maximizes $H(\pi, C)$, subject to $\Omega = \{(x, y) \in \mathcal{X} \times \mathcal{Y} \mid x \in y\}$

- Order $\mathcal{X} = \{x_1, \ldots, x_n\}$ such that $\pi(x_1) \geq \cdots \geq \pi(x_n)$
- Build C by mapping $x_1, \ldots, x_k$ to one output, $x_{k+1}, \ldots, x_{2k}$ to another and so on.

A greedy solution

The k-hypergraph Problem

Let $\mathcal{Y} = \{\mathcal{A} \subset \mathcal{X} \mid |\mathcal{A}| \leq k\}$. Find $C : \mathcal{X} \to \mathcal{Y}$ that maximizes $H(\pi, C)$, subject to $\Omega = \{(x, y) \in \mathcal{X} \times \mathcal{Y} \mid x \in y\}$

- Order $\mathcal{X} = \{x_1, \ldots, x_n\}$ such that $\pi(x_1) \geq \cdots \geq \pi(x_n)$
- Build C by mapping $x_1, \ldots, x_k$ to one output, $x_{k+1}, \ldots, x_{2k}$ to another and so on.

Greedy solution for 8 secret values, and $k = 3$

π
0.25
0.20
0.15
0.13
0.10
0.08
0.07
0.02

C	y_1	y_2	y_3
x_1	1	0	0
x_2	1	0	0
x_3	1	0	0
x_4	0	1	0
x_5	0	1	0
x_6	0	1	0
x_7	0	0	1
x_8	0	0	1

A greedy solution

Theorem

For leakage-supermodular entropies, the greedy solution is optimal

A greedy solution

Theorem

For leakage-supermodular entropies, the greedy solution is optimal

Proof idea.

π
0.3
0.3
0.2
0.1
0.1

C	y_1	y_2	y_3
x_1	1	0	0
x_2	0	1	0
x_3	1	0	0
x_3	0	1	0
x_3	0	0	1

A greedy solution

Theorem

For leakage-supermodular entropies, the greedy solution is optimal

Proof idea.

p	y_1	y_2	y_3
x_1	0.3	0	0
x_2	0	0.3	0
x_3	0.2	0	0
x_3	0	0.1	0
x_3	0	0	0.1

A greedy solution

Theorem

For leakage-supermodular entropies, the greedy solution is optimal

Proof idea. $H(\pi, C) = \eta\big(G_F(0.3, 0.2) + G_F(0.3, 0.1) + G_F(0.1)\big)$

p	y_1	y_2	y_3
x_1	0.3	0	0
x_2	0	0.3	0
x_3	0.2	0	0
x_3	0	0.1	0
x_3	0	0	0.1

A greedy solution

Theorem

For leakage-supermodular entropies, the greedy solution is optimal

Proof idea. $H(\pi, C) = \eta\big(G_F(0.3, 0.2) + G_F(0.3, 0.1) + G_F(0.1)\big)$

p	y_1	y_2	y_3
x_1	0.3	0	0
x_2	0	0.3	0
x_3	0.2	0	0
x_3	0	0.1	0
x_3	0	0	0.1

A greedy solution

Theorem

For leakage-supermodular entropies, the greedy solution is optimal

Proof idea. $H(\pi, C) = \eta\big(G_F(0.3, 0.2) + G_F(0.3, 0.1) + G_F(0.1)\big)$

p'	y_1	y_2	y_3
x_1	0.3	0	0
x_2	0.3	0	0
x_3	0	0.2	0
x_3	0	0.1	0
x_3	0	0	0.1

A greedy solution

Theorem

For leakage-supermodular entropies, the greedy solution is optimal

Proof idea. $H(\pi, C) = \eta\big(G_F(0.3, 0.2) + G_F(0.3, 0.1) + G_F(0.1)\big)$

p'	y_1	y_2	y_3
x_1	0.3	0	0
x_2	0.3	0	0
x_3	0	0.2	0
x_3	0	0.1	0
x_3	0	0	0.1

A greedy solution

Theorem

For leakage-supermodular entropies, the greedy solution is optimal

Proof idea. $H(\pi, C) = \eta\big(G_F(0.3, 0.2) + G_F(0.3, 0.1) + G_F(0.1)\big)$

π
0.3
0.3
0.2
0.1
0.1

C'	y_1	y_2	y_3
x_1	1	0	0
x_2	1	0	0
x_3	0	1	0
x_3	0	1	0
x_3	0	0	1

$$H(\pi, C') = \eta\big(G_F(0.3, 0.3) + G_F(0.2, 0.1) + G_F(0.1)\big)$$

A greedy solution

Theorem

For leakage-supermodular entropies, the greedy solution is optimal

Proof idea. $H(\pi, C) = \eta\big(G_F(0.3, 0.2) + G_F(0.3, 0.1) + G_F(0.1)\big)$

π
0.3
0.3
0.2
0.1
0.1

C'	y_1	y_2	y_3
x_1	1	0	0
x_2	1	0	0
x_3	0	1	0
x_3	0	1	0
x_3	0	0	1

$$H(\pi, C') = \eta\big(G_F(0.3, 0.3) + G_F(0.2, 0.1) + G_F(0.1)\big)$$

$$(0.3, 0.3) = (0.3, 0.2) \vee (0.1, 0.3)$$

A greedy solution

Theorem

For leakage-supermodular entropies, the greedy solution is optimal

Proof idea. $H(\pi, C) = \eta\big(G_F(0.3, 0.2) + G_F(0.3, 0.1) + G_F(0.1)\big)$

π
0.3
0.3
0.2
0.1
0.1

C'	y_1	y_2	y_3
x_1	1	0	0
x_2	1	0	0
x_3	0	1	0
x_3	0	1	0
x_3	0	0	1

$$H(\pi, C') = \eta\big(G_F(0.3, 0.3) + G_F(0.2, 0.1) + G_F(0.1)\big)$$

$$(0.1, 0.2) = (0.3, 0.2) \wedge (0.1, 0.3)$$

A greedy solution

Theorem

For leakage-supermodular entropies, the greedy solution is optimal

Proof idea. $H(\pi, C) = \eta\big(G_F(0.3, 0.2) + G_F(0.3, 0.1) + G_F(0.1)\big)$

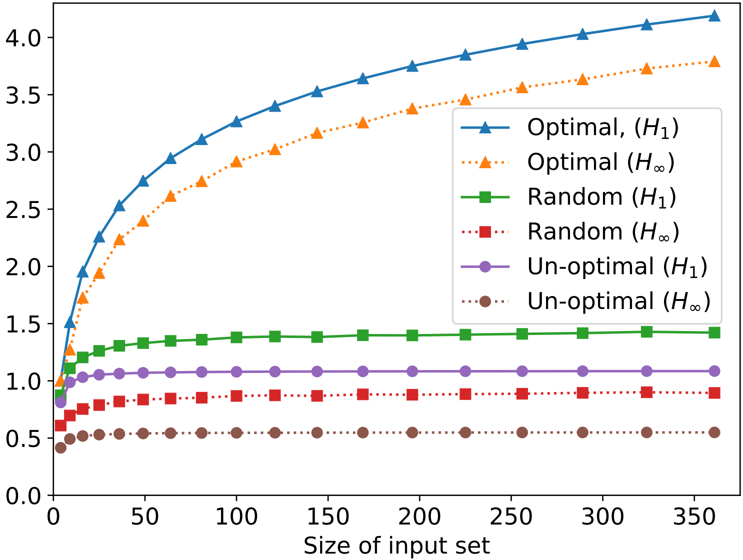
π
0.3
0.3
0.2
0.1
0.1

C'	y_1	y_2	y_3
x_1	1	0	0
x_2	1	0	0
x_3	0	1	0
x_3	0	1	0
x_3	0	0	1

$$H(\pi, C') = \eta\big(G_F(0.3, 0.3) + G_F(0.2, 0.1) + G_F(0.1)\big)$$

$$H(\pi, C') \geq H(\pi, C)$$

Experimental comparison



Deterministic Channels as a Solution for Multiple Executions

Deterministic Channels as a Solution for Multiple Executions

- Often, a system is executed multiple times for a fixed secret value

Deterministic Channels as a Solution for Multiple Executions

- Often, a system is executed multiple times for a fixed secret value
- How do we design an optimal system in this scenario?

Deterministic Channels as a Solution for Multiple Executions

C	y_1	y_2	y_3
x_1	?	?	0
x_2	?	0	?
x_3	0	?	?
x_4	?	0	?
x_5	?	?	?

Deterministic Channels as a Solution for Multiple Executions

C	y_1	y_2	y_3
x_1	$1/2$	$1/2$	0
x_2	$1/3$	0	$2/3$
x_3	0	$3/4$	$1/4$
x_4	$1/3$	0	$2/3$
x_5	$3/5$	$1/5$	$1/5$

Deterministic Channels as a Solution for Multiple Executions

C	y_1	y_2	y_3
x_1	$1/2$	$1/2$	0
x_2	$1/3$	0	$2/3$
x_3	0	$3/4$	$1/4$
x_4	$1/3$	0	$2/3$
x_5	$3/5$	$1/5$	$1/5$

$\longrightarrow$

C	$[x_1]$	$[x_2]$	$[x_3]$	$[x_5]$
x_1	1	0	0	0
x_2	0	1	0	0
x_3	0	0	1	0
x_4	0	1	0	0
x_5	0	0	0	1

Deterministic Channels as a Solution for Multiple Executions

C	y_1	y_2	y_3
x_1	0	1	0
x_2	0	0	1
x_3	0	1	0
x_4	0	0	1
x_5	0	1	0

C	$[x_1]$	$[x_2]$	$[x_3]$	$[x_5]$
x_1	1	0	0	0
x_2	0	1	0	0
x_3	0	0	1	0
x_4	0	1	0	0
x_5	0	0	0	1

Deterministic Channels as a Solution for Multiple Executions

Proposition

Let C be a probabilistic channel respecting the operational constraints. Then, there is a deterministic channel D that respects the same constraints and asymptotically leaks at most as much information as C

Deterministic Channels as a Solution for Multiple Executions

Proposition

Let C be a probabilistic channel respecting the operational constraints. Then, there is a deterministic channel D that respects the same constraints and asymptotically leaks at most as much information as C

Thus, the deterministic solution to the design problem is asymptotically optimal

Conclusions and Contributions

Conclusions and Contributions

In this work we...

Conclusions and Contributions

In this work we...

- Investigated the Deterministic Channel Design Problem: NP-hardness and non-universality

Conclusions and Contributions

In this work we...

- Investigated the Deterministic Channel Design Problem: NP-hardness and non-universality
- Established a greedy solution for the k-hypergraph problem which is optimal for the most common entropy measures

Conclusions and Contributions

In this work we...

- Investigated the Deterministic Channel Design Problem: NP-hardness and non-universality
- Established a greedy solution for the k-hypergraph problem which is optimal for the most common entropy measures
- Introduced leakage-supermodularity, which may be a useful concept for future work in QIF

Conclusions and Contributions

In this work we...

- Investigated the Deterministic Channel Design Problem: NP-hardness and non-universality
- Established a greedy solution for the k-hypergraph problem which is optimal for the most common entropy measures
- Introduced leakage-supermodularity, which may be a useful concept for future work in QIF
    - *Channel Ordering and Supermodularity* – to appear at IEEE ITW 2019

Conclusions and Contributions

In this work we...

- Investigated the Deterministic Channel Design Problem: NP-hardness and non-universality
- Established a greedy solution for the k-hypergraph problem which is optimal for the most common entropy measures
- Introduced leakage-supermodularity, which may be a useful concept for future work in QIF
  - *Channel Ordering and Supermodularity* – to appear at IEEE ITW 2019
- Proved that, if a system is to be executed multiple times, the deterministic solution is optimal when the number of executions is very large