IEEE WPS 2020 - International Workshop on Privacy and Security for Information Systems

Optimal Mechanisms Under Maximal Leakage

Benjamin Wu
Electrical and Computer Engineering
Cornell University
Ithaca, USA
bhw49@cornell.edu

Aaron B. Wagner
Electrical and Computer Engineering
Cornell University
Ithaca, USA
wagner@cornell.edu

G. Edward Suh
Electrical and Computer Engineering
Cornell University
Ithaca, USA
suh@ece.cornell.edu

***Abstract*—Side channels represent a broad class of security vulnerabilities in practical systems. Because completely eliminating side channels often leads to prohibitively high overhead, there is a need for principled techniques that trade off cost and leakage. Maximal leakage (MaxL) has been introduced as an operationally-interpretable leakage metric well-suited to side channels. We study the optimal trade-off between MaxL and expected costs. We demonstrate that for an important class of cost functions, optimal protection can be achieved using a combination of at most two deterministic schemes. We discuss the implications of this result for practical implementation and provide a fast heuristic algorithm for finding the best deterministic mechanism, which has a bounded suboptimality guarantee.**

I. Introduction

Side channels represent a broad class of security vulnerabilities that have received significant attention from the cybersecurity community, especially after the demonstration of multiple side channel-based attacks [4], [10], [12]. Recently, *maximal leakage* (MaxL) [3] was introduced as an operationally-interpretable measure of the usefulness of a side channel to an attacker. While quantifying the leakage from side channels is clearly of great interest, it alone is not sufficient for deployment in practical settings. We address a critical question that naturally arises, once armed with a metric such as MaxL. Namely, how can a security-minded system designer devise a protection scheme that minimizes MaxL, subject to a cost constraint (or vice versa)? Such trade-offs are certainly of interest, and tuneable protection schemes have already been studied to an extent for other leakage metrics[13], [1]. While some early attempts at studying MaxL in the context of cost-leakage analysis have been made [2], [6], [5], here we address the problem of formulating MaxL-based, tuneable protection schemes from the perspective of a system designer.

MaxL, in its most basic form, assumes the existence of three random variables in a side channel. Let U denote the random variable representing the victim's secret, X denote some intermediate value visible within the system but not to the adversary, and Y denote the value observed by the adversary. U, X, and Y form a Markov Chain (denoted as $U - X - Y$), so Y and U are conditionally independent given X. While continuous versions of MaxL exist, for the purposes of this study we will assume that X and Y are discrete random variables with finite alphabets $\mathcal{X}$ and $\mathcal{Y}$.

Given these random variables, MaxL is defined as [3]:

$$\mathscr{L}(X \to Y) = \max_{U:U-X-Y} \log \frac{\max_{\tilde{u}(\cdot)} P(U = \tilde{u}(Y))}{\max_{\tilde{u}} P(U = \tilde{u})} \qquad (1)$$

and also has a more easily computed, equivalent form [3]:

$$\mathscr{L}(X \to Y) = \log \sum_{y \in \mathcal{Y}} \max_{x \in \mathcal{X}} p(y|x) \qquad (2)$$

The focus of this paper is on the design of protection schemes (i.e., a conditional distribution of Y given X) that optimally trade off MaxL with cost. We assume the existence of a cost matrix (subject to a broadly applicable structural constraint that we will define later) that defines the one-time cost associated with mapping any particular element of $\mathcal{X}$ to any particular element of $\mathcal{Y}$, and consider the ensuing expected cost under the given protection scheme.

Our contributions in this paper are to show that: 1) the linear program (LP) that follows from the above is achieved by nearly deterministic protection schemes, a fact that has some useful implications for practical deployment, and 2) the entire cost-leakage trade-off curve can be derived by solving the LP at no more than $\min\{|\mathcal{X}|, |\mathcal{Y}|\}$ points, and 3) a greedy heuristic exists for approximating these points. Due to space constraints, we omit discussion of related leakage metrics, but we point out that metrics such as mutual information or differential privacy do not have these properties. A more complete comparison with other metrics is available in the extended version of the paper [11].

II. Definitions and Cost Functions

In this section, we will define some necessary terms and variables, as well as clarify the class of costs under consideration.

Definition 1. ***(Basic Notation)*** *For X and Y as previously defined, with alphabet sizes $|\mathcal{X}| = M$ and $|\mathcal{Y}| = N$, we define the following:*

- *$c(x, y)$ is the nonnegative (but not necessarily finite) cost of mapping each $x \in \mathcal{X}$ to each $y \in \mathcal{Y}$. We refer to this function as the* cost function *and the corresponding matrix $\{c_{xy}\}$ as the* cost matrix.
- *For any matrix $\boldsymbol{A} = \{a_{xy}\}$, $\mathscr{C}(\boldsymbol{A}) = \sum_{x,y} p(x)c(x,y)a_{xy}$ is the* total cost *of $\boldsymbol{A}$.*
- *For any matrix $\boldsymbol{A} = \{a_{xy}\}$, $\mathscr{L}(\boldsymbol{A}) = \sum_y \max_x a_{xy}$ is the* exponentiated maximal leakage *(or* exp-leak, *for*

978-1-7281-4760-4/20/$31.00 ©2020 IEEE

short) of $\boldsymbol{A}$*. Note that minimizing over exp-leak is equivalent to minimizing over maximal leakage.*

- *Any* $M \times N$ *transition matrix (rows sum to 1, non-negative entries)* $\boldsymbol{P} = \{p_{xy}\}$*, such that* $p_{xy} = P(Y = y|X = x)\ \ \forall x, y$ *and* $\mathscr{C}(\boldsymbol{P})$ *is finite, is called a* protection scheme.

Definition 2. ***(Specialized Terminology)***

- *A protection scheme* $\boldsymbol{P}$ *is* deterministic *if all* p_{xy} *equal 0 or 1. It is* stochastic *otherwise.*
- *The ordered pair* $(L, C) \in \mathbb{R}^2$ *is* achieved *by protection scheme* $\boldsymbol{P}$ *if* $\mathscr{L}(\boldsymbol{P}) \le L$ *and* $\mathscr{C}(\boldsymbol{P}) \le C$.
- *The ordered pair* (L, C) *is* achievable *if there exists protection scheme* $\boldsymbol{P}$ *such that* (L, C) *is achieved by it.*
- *The set* S *is the set of all achievable* (L, C) *pairs.*
- $C^*(L) = \inf\,[C : (L, C) \in S]$*. We refer to* $C^*(L)$ *evaluated for all values of* L *as the* trade-off curve *and the set of points* $S_b = [(L, C) \in S | C = C^*(L)]$ *as the* boundary *of* S.
- $\boldsymbol{P}$ *is* optimizing in S *if* $\mathscr{C}(\boldsymbol{P}) = C^*(\mathscr{L}(\boldsymbol{P}))$ *(i.e. if* $\boldsymbol{P}$ *achieves a point on the boundary of* S*).*
- *The set* S_d *is the set of all points in* S *that can be achieved by a deterministic protection scheme.*
- *The ordered pair* (L, C) *is* achievable in S_d *if there exists a deterministic protection scheme* $\boldsymbol{P}$ *that achieves* (L, C).
- $C_d^*(L) = \inf\,[C : (L, C) \in S_d]$.
- $\boldsymbol{P}$ *is* optimizing in S_d *if* $\mathscr{C}(\boldsymbol{P}) = C_d^*(\mathscr{L}(\boldsymbol{P}))$*. Note that a* $\boldsymbol{P}$ *that is optimizing in* S_d *is not necessarily a deterministic protection scheme.*

Finally, we discuss the aforementioned constraints on the cost matrix that we will work with in this study:

Definition 3. ***(Staircase nondecreasing cost functions)***

For $|\mathcal{X}| = M$ *and* $|\mathcal{Y}| = N$*, let* $\mathcal{X} = \{x_1, x_2, ...x_M\}$ *and* $\mathcal{Y} = \{y_1, y_2, ...y_N\}$*. We refer to a cost function/matrix that satisfies the following constraints as* staircase nondecreasing*:*

1) *For* $0 < i < j \le M$ *and all* $y \in \mathcal{Y}$*, if* $c(x_i, y) = \infty$*, then* $c(x_j, y) = \infty$*. (i.e. if one matrix element is infinite, then that column is infinite all the way down).*
2) *For* $0 < i < j \le N$ *and all* $x \in \mathcal{X}$*, if* $c(x, y_i) < \infty$*, then* $c(x, y_i) \le c(x, y_j) < \infty$*. (i.e. excluding infinities, each row of the matrix is nondecreasing from left to right).*

Note that staircase nondecreasing cost matrices are exemplified by upper triangular cost matrices with ordered cost entries for each row, a special case that is typical of most power and timing side channels due to causality constraints.

III. Main Result

We consider the minimization of total cost subject to an MaxL constraint, written as the following LP using standard techniques (let q_y denote the column maxima)

$$\begin{aligned} C^*(L) = \min_{p_{xy}, q_y} \mathscr{C}(\mathbf{P}) \quad \text{s.t.} \sum_y q_y \le L,\ \sum_y p_{xy} = 1\ \forall x, \\ p_{xy} \ge 0,\ p_{xy} \le q_y,\ \forall\ x, y \end{aligned} \tag{3}$$

Remark. *$C^*(L)$ is a convex function of L. The proof follows from standard arguments.*

Theorem 1. ***(Main Theorem)***

If $c(x, y)$ *is staircase nondecreasing, then*

1) $\min\limits_{(\mathscr{L},\mathscr{C}) \in S} \mathscr{C} + \alpha\mathscr{L} = \min\limits_{(\mathscr{L},\mathscr{C}) \in S_d} \mathscr{C} + \alpha\mathscr{L} \quad \forall\alpha > 0$
2) *For all* $L \ge 1$*,* $(L, C^*(L))$ *can be achieved by* $\boldsymbol{P} = \lambda\boldsymbol{P}_1 + (1 - \lambda)\boldsymbol{P}_2$ *for some* $\lambda \in [0, 1]$ *and some deterministic protection schemes* $\boldsymbol{P}_1$ *and* $\boldsymbol{P}_2$*, such that* $\mathscr{L}(\boldsymbol{P}) \le L$ *and* $C^*(L) \le \lambda C_d^*(\mathscr{L}(\boldsymbol{P}_1)) + (1 - \lambda)C_d^*(\mathscr{L}(\boldsymbol{P}_2))$.

The proof proceeds as follows. First, for any protection scheme, we will define its *water-filled* form and show that for any staircase nondecreasing cost function, an optimizing protection scheme's water-filled form is also optimizing. Second, we will show that any water-filled protection scheme can be transformed in such a way that it progressively approaches a deterministic protection scheme while remaining optimizing. The first part of the theorem will follow by iteratively applying this transformation on an optimizing protection scheme a finite number of times. The second part of the theorem will follows using standard convex analysis.

A. Proof of Theorem 1.1

Lemma 2. ***(Water-Filling Lemma)***

Consider any protection scheme $\boldsymbol{P}$*. Define a* $1 \times N$ *vector* $\vec{p} = [p_1, p_2, ...p_N]$ *such that* $p_i = \max_{x \in \mathcal{X}} p_{xy_i}$ *(i.e.,* $\vec{p}$ *consists of the column maxima of* $\boldsymbol{P}$*). Using* $\vec{p}$ *alone, we construct a new protection scheme* **P'** *as follows:*

1) *Start with a* $M \times N$ *zero matrix* **P'** $= \{p'_{xy}\}$.
2) *For each row* $x \in \mathcal{X}$*, iterate over each* y_i*,* $i = 1, 2, ...N$.
   - *If* $c(x, y_i) = \infty$*, let* $p'_{xy_i} = 0$
   - *Else, set* $p'_{xy_i} = \min\{p_i, 1 - \sum_{j=1}^{i-1} p'_{xy_j}\}$.

In plain terms, we are constructing **P'** *by maintaining the column maxima of* **P** *and "filling" in probability mass in each row from left to right. We call* **P'** *the water-filled form of* **P***. Also, if* **P** *and* **P'** *are identical, we say that* **P** *is a water-filled protection scheme.*

Then, if the cost function satisfies definition 3, all optimizing **P** *can be converted into water-filled form* **P'** *such that* $\mathscr{C}(\mathbf{P}) = \mathscr{C}(\mathbf{P'})$ *and* $\mathscr{L}(\mathbf{P}) = \mathscr{L}(\mathbf{P'})$.

Proof of Lemma 2. Suppose we are given optimizing **P** and its water-filled form **P'**. By its construction, $\mathscr{L}(\mathbf{P}) \ge \mathscr{L}(\mathbf{P'})$ since we did not increase the total sum of column maxima. In addition, since we independently fill up each row's entries in **P'** from least cost to greatest cost, $\mathscr{C}(\mathbf{P}) \ge \mathscr{C}(\mathbf{P'})$ for any cost function that is staircase nondecreasing. It then follows that: $\mathscr{L}(\mathbf{P}) \leq \mathscr{L}(\mathbf{P'})$ and $\mathscr{C}(\mathbf{P}) \leq \mathscr{C}(\mathbf{P'})$, since $\mathbf{P}$ is optimizing so $\mathscr{C}(\mathbf{P}) + \alpha\mathscr{L}(\mathbf{P}) \leq \mathscr{C}(\mathbf{P'}) + \alpha\mathscr{L}(\mathbf{P'})$. Therefore, $\mathscr{C}(\mathbf{P}) = \mathscr{C}(\mathbf{P'})$ and $\mathscr{L}(\mathbf{P}) = \mathscr{L}(\mathbf{P'})$. □

For the rest of the proof of Theorem 1.1, for any optimizing $\mathbf{P}$, we assume it is already in water-filled form, since we have already shown that doing so does not unnecessarily restrict our space of optimizing solutions.

Now, we would like to show that there exists a special choice of $\mathbf{Q}$ such that $\mathscr{C}(\mathbf{P} + \delta\mathbf{Q}) + \alpha\mathscr{L}(\mathbf{P} + \delta\mathbf{Q})$:

1) is linear over some well-defined interval of δ values around 0 (*linearity*)
2) does not vary with δ for any fixed α (*no improvement* with δ)
3) results in protection scheme $\mathbf{P} + \delta\mathbf{Q}$ being strictly "more deterministic" (to be defined shortly) than $\mathbf{P}$ for a particular choice of δ (*more deterministic*)

as doing so will allow us to transform $\mathbf{P}$ into an optimizing deterministic protection scheme by iteratively choosing different $\mathbf{Q}$'s and applying the above transformation.

Definition 4. ***(Types of Matrix Entries)***

For the sake of discourse, we will define the following types of matrix entries in any protection scheme:

- *An entry is* fractional *if it is not equal to 0 or 1, and* integral *otherwise. Similarly, a column is fractional if its maximum entry is fractional and integral otherwise.*
- *An entry is* maxed out *if it is equal to the maximum value in its column, and* hanging *otherwise.*

Note that it is true by construction that a water-filled protection scheme will have at most one hanging mass entry and at least one maxed out entry in each row. Moreover, if a row has a hanging mass entry, there do not exist other non-zero entries further to the right of that entry.

Definition 5. ***(Measure of Randomness)***

$R(\boldsymbol{P})$ = (# fractional columns in $\boldsymbol{P}$) + (# hanging entries in $\boldsymbol{P}$)

Note that $R(\boldsymbol{P}) = 0$ if and only if $\boldsymbol{P}$ is a deterministic protection scheme. $R(\boldsymbol{P})$ should be thought of as a measure of how stochastic a protection scheme is.

We first propose a particular choice of $\mathbf{Q}$ and δ.

Definition 6. ***(Q-Generation Procedure)***

Given any water-filled protection scheme $\boldsymbol{P}$ with at least one fractional entry, we now give a procedure to generate a $\boldsymbol{Q}$ matrix. Note that any such protection scheme must also have at least one fractional column or else it would contradict the water-filled property.

1) *Start with an $M \times N$ zero matrix $\boldsymbol{Q}$ that we will populate with values.*
2) *Denote the leftmost fractional column index in $\boldsymbol{P}$ as y. Further denote the current "sign" to "+".*
3) *In the yth column of $\boldsymbol{Q}$, if the sign is "+", assign the value 1 to all entries in that column that are maxed out in $\boldsymbol{P}$. If the sign is "-", assign the value -1 instead.*
4) *If the current sign is "+", change it to "-", and vice versa.*
5) *Consider the set of rows that are maxed out in the yth column of $\boldsymbol{P}$. Do all of these rows either have hanging mass in $\boldsymbol{P}$ or already have 2 non-zero entries in $\boldsymbol{Q}$? If so, go to step 9; otherwise proceed to step 6.*
6) *Again consider the set of rows that are maxed out in the y^{th} column of $\boldsymbol{P}$. Choose the topmost row from this set that does not have hanging mass in $\boldsymbol{P}$ and has only 1 non-zero entry in $\boldsymbol{Q}$. Denote the row index of that entry as x.*
7) *Set y to be the column index of the rightmost, maxed out entry of the xth row in the $\boldsymbol{P}$ matrix. Note that y must correspond to a fractional column here.*
8) *Go to step 3.*
9) *If any rows in $\boldsymbol{Q}$ have hanging mass and an odd number of non-zero entries, assign either 1 or -1, so that each of these rows sum to 0, to the hanging mass entries of these rows.*

Lemma 3. ***(Q-Generation Properties)*** *The procedure specified by definition 6 satisfies the following:*

1) *The procedure terminates.*
2) *All of the rows in the generated $\boldsymbol{Q}$ matrix sum to 0 (so that $\boldsymbol{P} + \delta\boldsymbol{Q}$ is a protection scheme).*
3) *$\boldsymbol{P} + \delta\boldsymbol{Q}$ is a water-filled protection scheme*

Proof of Lemma 3.1. Since we never choose columns that are not fractional, any row selected in step 6 must have a maxed out entry (because we also ignore rows with fractional entries) somewhere to the right of the current y column. Certainly, this procedure must terminate if the y value ever reaches the right-most column (and the process may terminate earlier than that due to step 5). □

Proof of Lemma 3.2. Since we only assign 1 and -1 to entries of $\mathbf{Q}$ in alternation, this is the same as saying that each row must contain an even number of non-zero entries. We see that this is true by noting that there are three types of rows, differentiated by how their non-zero entries in $\mathbf{Q}$ (if any) are assigned during the Q-generating procedure.

If a row has hanging mass in $\mathbf{P}$, then step 9 will necessarily adjust that row to have an even number of non-zero entries by construction. In addition, we never assign mass to hanging mass entries until step 9, when the procedure terminates, which means that all hanging mass entries are free for us to use at that point. So, rows that have hanging mass in $\mathbf{P}$ will be valid rows in $\mathbf{Q}$.

If a row has no hanging mass in $\mathbf{P}$, then there are two cases, depending on whether that row was ever used in step 6 to determine the next y value (we will refer to such a row as "critical"). Note that, due to steps 5 and 6 filtering out rows that already have 2 non-zero entries, no row will ever be used in step 6 twice (i.e. a row will be a critical row at most once).

If the row is critical, it must be the topmost one that had only one non-zero entry in $\mathbf{Q}$ at that point of the procedure in the previous yth column. Step 7 guarantees that the only other non-zero entry in this row will correspond to its rightmost non-zero entry in $\mathbf{P}$. So this row will have exactly 2 non-zero entries in $\mathbf{Q}$, making it valid.

If the row is not critical, it must either be located below one that is or not have any non-zero entries in $\mathbf{Q}$ at all. The latter case results in a trivially valid row. In the former case, the row must have at least two non-zero entries in columns shared with the previous critical row, or else it would violate our assumptions that $\mathbf{P}$ is water-filled and the cost function is staircase nondecreasing. In addition, since $\mathbf{P}$ is water-filled, each row is majorized by all rows above it (i.e. the cumulative left-to-right sum of the upper row is no less than that of the lower row for every column). This implies that a non-critical row cannot have more than 2 non-zero entries either. □

Proof of Lemma 3.3. We observe that due to step 3, we only ever change all of the maxed out entries in a column together. So, for small δ, $\mathbf{P}+\delta\mathbf{Q}$ will remain water-filled. □

At this point, we will show that $\mathscr{C}(\mathbf{P}+\delta\mathbf{Q})+\alpha\mathscr{L}(\mathbf{P}+\delta\mathbf{Q})$ has the aforementioned properties of linearity, no improvement with δ, and is more deterministic.

Definition 7. ***(Stopping Conditions)***

Let $\delta_+ = \sup[\delta \geq 0 : \boldsymbol{P}+\delta\boldsymbol{Q}$ is stochastic and $\boldsymbol{P}$ and $\boldsymbol{P}+\delta\boldsymbol{Q}$ are maxed out for the same entries and fractional for the same entries$]$

and $\delta_- = \inf[\delta \leq 0 : \boldsymbol{P}+\delta\boldsymbol{Q}$ is stochastic and $\boldsymbol{P}$ and $\boldsymbol{P}+\delta\boldsymbol{Q}$ are maxed out for the same entries and fractional for the same entries$]$

Note that, by definition $\delta_+ > 0$ and $\delta_- < 0$.

Lemma 4. ***(Linearity Lemma)***

If $\boldsymbol{P}$ is water-filled for fixed α and $\boldsymbol{Q}$ is generated according to definition 6, then $\mathscr{C}+\alpha\mathscr{L}$ evaluated with $\boldsymbol{P}+\delta\boldsymbol{Q}$ is linear with respect to $\delta \in [\delta_-, \delta_+]$.

Proof of Lemma 4. For $\delta_- < \delta < \delta_+$ and fixed α,

$$
\begin{aligned}
&\mathscr{C}(\mathbf{P}+\delta\mathbf{Q})+\alpha\mathscr{L}(\mathbf{P}+\delta\mathbf{Q})\\
&=\sum_x\sum_y p(x)c(x,y)(p_{xy}+\delta q_{xy})+\alpha\sum_y \max_x(p_{xy}+\alpha q_{xy})\\
&=\sum_x\sum_y p(x)c(x,y)(p_{xy}+\delta q_{xy})+\alpha\sum_y (p_{x(y)y}+\alpha q_{x(y)y})
\end{aligned}
$$

where $x(y)=\arg\max_x p_{xy}$.

Since $\mathscr{C}(\mathbf{P}+\delta\mathbf{Q})+\alpha\mathscr{L}(\mathbf{P}+\delta\mathbf{Q})$ is linear over (δ_-, δ_+) and continuous over $[\delta_-, \delta_+]$, it is linear over $[\delta_-, \delta_+]$. □

Lemma 5. ***(No Improvement Lemma)***

If $\boldsymbol{P}$ minimizes $\mathscr{C}+\alpha\mathscr{L}$ over S for fixed α and is water-filled and $\boldsymbol{Q}$ is generated according to definition 6, then $\frac{\partial}{\partial\delta}(\mathscr{C}+\alpha\mathscr{L})=0$ at $\delta=0$.

Proof of Lemma 5. If $\frac{\partial}{\partial\delta}(\mathscr{C}+\alpha\mathscr{L})\neq 0$, then that implies that $\mathbf{P}+\delta\mathbf{Q}$ performs strictly better for some δ close to zero, which is a contradiction. □

Lemma 6. ***(More Deterministic Lemma)***

If $\boldsymbol{P}$ is water-filled for fixed α and $\boldsymbol{Q}$ is generated according to Definition 6, then $R(\boldsymbol{P}+\delta\boldsymbol{Q}) < R(\boldsymbol{P})$ for both $\delta=\delta_-$ or $\delta=\delta_+$ as defined by Definition 7.

Proof of Lemma 6. As δ increases from 0 to δ_+, some fractional entries of $\mathbf{P}+\delta\mathbf{Q}$ change, and none of the integral entries change. In addition, if one maxed out entry changes, all of the maxed out entries in that column change together. It thus follows that the set of fractional columns can only decrease with δ and that the set of hanging entries likewise can only decrease. So $R(\mathbf{P})$ is nonincreasing in δ for $\delta\in[0,\delta_+]$. From the definition of δ_+ in Definition 7, $R(\mathbf{P}+\delta_+\mathbf{Q}) < R(\mathbf{P})$.

Similarly, we can show that $R(\mathbf{P}+\delta_-\mathbf{Q}) < R(\mathbf{P})$. □

Proof of Theorem 1.1. Any $\mathbf{P}$ that minimizes $\mathscr{C}+\alpha\mathscr{L}$ for some α can be chosen to be optimizing and water-filled as per Lemma 2. If $\mathbf{P}$ is not a deterministic protection scheme, we can select $\mathbf{Q}$ as in Definition 6 with the properties shown in Lemma 3.

By Lemmas 4, 5, 6, we know $\mathscr{C}(\mathbf{P}+\delta\mathbf{Q})+\alpha\mathscr{L}(\mathbf{P}+\delta\mathbf{Q})$ is constant over $[\delta_-, \delta_+]$ and $R(\mathbf{P}+\delta\mathbf{Q}) < R(\mathbf{P})$.

If $\mathbf{P}+\delta\mathbf{Q}$ is not deterministic, then we can repeat the above process since it is still water-filled and minimizes $\mathscr{C}+\alpha\mathscr{L}$ for the same α.

Eventually, after repeating this process some finite number of times, $R(\mathbf{P})$ will be 0 (since the function we defined is always nonnegative), and therefore deterministic. □

B. Proof of Theorem 1.2

Theorem 1.2. Using standard convex analysis (e.g. [8], chapter 12), Theorem 1.1 implies that that $C^*(L)$ and $C_d^*(L)$ have the same lower semi-continuous hull (or the closure, as defined by [8] chapter 7), which is equivalent to our definition of the boundary of S. We can see this fact as follows:

First, we note that the left and right hand sides of the equality in Theorem 1.1 are the conjugate functions of $C^*(L)$ and $C_d^*(L)$, respectively. We have shown that the conjugates are equal for any α.

Second, since $C^*(L)$ is a convex function of L, the conjugate of the conjugate of $C^*(L)$ is equal to the closure of $C^*(L)$ ([8], Corollary 13.1.1).

Third, while $C_d^*(L)$ is not a convex function, its conjugate is the same as the conjugate of the closure of its convex hull. Therefore, the conjugate of its conjugate must be equal to the closure of its convex hull.

Thus, we have shown that the convex hulls of S and S_d are the same, since the two sets are the epigraphs of (all points in $\mathbb{R}^2$ on or above the curves defined by) the functions $C^*(L)$ and $C_d^*(L)$, respectively.

From this fact, it trivially follows that any (L, C) pair on the boundary of S must also lie on the convex hull of S, and therefore on the convex hull of S_d.

Finally, as previously noted, $C_d^*(L)$ is a descending staircase-like function for $L\in[1,\infty]$. So, the convex hull of S_d is given by the largest convex linear interpolation of the outer corner points of $C_d^*(L)$ (for example, see figure)

Therefore, any (L, C) pair on the boundary of S is achievable by a convex combination of no more than two deterministic protection schemes. □

C. Discussion of Theorem 1

There are three practical implications of the main theorem.

First, any deterministic protection scheme can be compressed to an $N \times 2$ (or smaller) look-up table, in contrast to the $N \times M$ table required for general stochastic schemes. Deterministic schemes also naturally do not require the generation of randomness. If a mixture of two deterministic schemes is needed, one may implement a pre-determined schedule alternating between the two deterministic schemes. In addition, deterministic schemes are resistant to averaging attacks, where the adversary attempts to learn additional information by gathering statistics of Y, since the same X value always maps the the same Y value.

Note that mechanisms designed to minimize mutual information tend to be highly stochastic. As an example, consider the alphabets $\mathcal{X} = \{x_1, x_2, x_3, x_4\}$ and $\mathcal{Y} = \{y_1, y_2, y_3, y_4\}$, the marginal distribution of X, $p(x) = [0.4, 0.2, 0.2, 0.2]$, and the cost function

$$C = \begin{bmatrix} 1 & 2 & 3 & 4 \\ \infty & 1 & 2 & 3 \\ \infty & \infty & 1 & 2 \\ \infty & \infty & \infty & 1 \end{bmatrix}$$

the MaxL-optimal and MI-optimal solutions for 0.5 units of cost are given by:

$$P^*_{ML} = \begin{bmatrix} .25 & .75 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P^*_{MI} = \begin{bmatrix} .5235 & .3031 & 0.1233 & 0.0502 \\ 0 & .4890 & .3120 & .1990 \\ 0 & 0 & .6105 & .3895 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Second, the proof of the main theorem provides an algorithm by which one may take any known optimizing protection scheme and convert it to a deterministic form that is also optimizing. This algorithm simply performs the procedures specified in Definitions 2 and 6 recursively.

Third, if it is necessary to solve the entire optimal trade-off curve (for example, if on-the-fly tuning of leakage is expected), it is only necessary to solve for integer exp-leak points and then connect the dots so that the overall curve is convex.

IV. A Heuristic Algorithm

As we just noted, if the full trade-off curve and the protection schemes are needed, then the entire curve can be computed by solving the LP at only the integer exp-leak points. Depending on the side channel, solving LPs for all of these points may be resource intensive. In this section, we present a fast heuristic with a bounded gap from optimality that can be used to approximate these points.

A. Greedy Algorithm

Definition 8. *For any nonempty set $\mathcal{S} \subseteq \mathcal{Y}$ and cost matrix $\{c(x, y)\}$, let $\boldsymbol{P}_\mathcal{S} = \{p_{xy}\}$ such that:*

$$p_{xy} = \begin{cases} 1 & \text{if } y = \min \arg\min_{y' \in \mathcal{S}} c(x, y') \\ 0 & \text{otherwise} \end{cases} \qquad (4)$$

We refer to $\boldsymbol{P}_\mathcal{S}$ as the deterministic protection scheme induced by *the subset $\mathcal{S}$.*

Definition 9. *For any non-empty set $\mathcal{S} \subseteq \mathcal{Y}$, let:*

$$\mathscr{L}(\mathcal{S}) = \mathscr{L}(\boldsymbol{P}_\mathcal{S}) \text{ and } \mathscr{C}(\mathcal{S}) = \mathscr{C}(\boldsymbol{P}_\mathcal{S}) \qquad (5)$$

Definition 10. *For a given staircase nondecreasing cost matrix $\{c(x, y)\}$, we identify one (not necessarily unique) $y_0 \in \mathcal{Y}$ such that:*

$$y_0 = \arg\min_{y \in \mathcal{Y}} \mathscr{C}(\{y\}) \qquad (6)$$

Define the subset $\mathcal{Y}' = \mathcal{Y} - \{y_0\}$.

Definition 11. *For any set $\mathcal{A} \subseteq \mathcal{Y}'$, we define the set function:*

$$f(\mathcal{A}) = -\mathscr{C}(\mathcal{A} \cup \{y_0\}) \qquad (7)$$

Definition 12. ***(Greedy Algorithm)***

1) *Start with $\mathcal{A} = \{\emptyset\}$.*
2) *Choose $y \in \mathcal{Y}' - \mathcal{A}$ such that $f(\mathcal{A} \cup \{y\})$ is maximized over all such choices of y. If $\mathcal{Y}' - \mathcal{A}$ is empty or if there does not exist such y that $f(\mathcal{A} \cup \{y\}) > f(\mathcal{A})$, terminate this algorithm.*
3) *Set $\mathcal{A} = \mathcal{A} \cup \{y\}$.*
4) *Go to step 2.*

B. Bounded Sub-optimality of the Greedy Algorithm

Using standard results in combinatorial optimization [7], we can obtain bounds on how suboptimal the solutions obtained from the greedy algorithm are. We will first prove some basic facts about the set function $f(\mathcal{A})$ given in Definition 11.

Lemma 7. *$f(\mathcal{A})$ is submodular.*

Proof. For $\mathcal{A}, \mathcal{B} \subseteq \mathcal{Y}'$ such that $\mathcal{A} \cap \mathcal{B} = \{\emptyset\}$,

$$\begin{aligned} f(\mathcal{A} \cup \mathcal{B}) &= -\sum_{x \in \mathcal{X}} \min_{y \in \mathcal{A} \cup \mathcal{B} \cup \{y_0\}} p(x) c(x, y) \\ &= -\sum_{x \in \mathcal{X}} \min_{y \in \mathcal{A} \cup \{y_0\}} p(x) c(x, y) + \sum_{x \in \mathcal{X}} \min_{y \in \mathcal{A} \cup \{y_0\}} p(x) c(x, y) \\ &\quad - \sum_{x \in \mathcal{X}} \min_{y \in \mathcal{A} \cup \mathcal{B} \cup \{y_0\}} p(x) c(x, y) \\ &= f(\mathcal{A}) + \sum_{x \in \mathcal{X}} p(x) [\min_{y \in \mathcal{A} \cup \{y_0\}} c(x, y) - \min_{y \in \mathcal{A} \cup \mathcal{B} \cup \{y_0\}} c(x, y)] \\ &\equiv f(\mathcal{A}) + D(A, B) \end{aligned}$$

Then, for $\mathcal{A} \subseteq \mathcal{Y}'$ and $b, c \in \mathcal{Y}' \backslash \mathcal{A}$,

$$\begin{aligned}
&f(\mathcal{A} \cup \{b\}) + f(\mathcal{A} \cup \{c\}) - f(\mathcal{A} \cup \{b, c\}) - f(\mathcal{A}) \\
&= D(\mathcal{A}, \{b\}) + D(\mathcal{A}, \{c\}) - D(\mathcal{A}, \{b, c\}) \\
&= \sum_{x \in \mathcal{X}} p(x) [\min_{y \in \mathcal{A} \cup \{y_0\}} c(x, y) - \min_{y \in \mathcal{A} \cup \{b, y_0\}} c(x, y) \\
&\qquad - \min_{y \in \mathcal{A} \cup \{c, y_0\}} c(x, y) + \min_{y \in \mathcal{A} \cup \{b, c, y_0\}} c(x, y)] \\
&\equiv \sum_{x \in \mathcal{X}} p(x) [C_1 - C_2 - C_3 + C_4] \geq 0
\end{aligned}$$

since C_4 is equal to C_2 or C_3 (or both), and C_1 is no smaller than either C_2 or C_3. Hence,

$$f(\mathcal{A} \cup \{b\}) + f(\mathcal{A} \cup \{c\}) \geq f(\mathcal{A} \cup \{b, c\}) + f(\mathcal{A}) \qquad (8)$$

so $f(\mathcal{A})$ is submodular ([9], Thm 44.1).

□

Definition 13. *For integer exp-leak bound L, let $\mathcal{A}_g(L)$ be the set obtained by running the greedy algorithm unil $|\mathcal{A} \cup \{y_0\}| = L$ (for simplicity, assume the greedy algorithm does not terminate prior to this point).*

For integer exp-leak bound L, let $\mathcal{A}^*(L) \subseteq \mathcal{Y}'$ be the true optimal set such that $f(\mathcal{A})$ is maximized subject to $|\mathcal{A} \cup \{y_0\}| \leq L$.

Now, since $f(\mathcal{A})$ is submodular, we can bound the greedy algorithm for all $L \geq 2$ as follows ([7], Theorem 4.1):

$$\begin{aligned}
&\frac{f(\mathcal{A}^*(L)) - f(\mathcal{A}_g(L))}{f(\mathcal{A}^*(L)) - f(\{\emptyset\})} \\
&= \frac{\mathscr{C}(\mathcal{A}_g(L) \cup \{y_0\}) - \mathscr{C}(\mathcal{A}^*(L) \cup \{y_0\})}{\mathscr{C}(\{y_0\}) - \mathscr{C}(\mathcal{A}^*(L) \cup \{y_0\})} \leq \Big(\frac{L-2}{L-1}\Big)^{L-1} \\
&\leq \frac{1}{e}
\end{aligned} \qquad (9)$$

References

[1] Gong, X., Kiyavash, N.: Quantifying the Information Leakage in Timing Side Channels in Deterministic Work-Conserving Schedulers. IEEE/ACM Transactions on Networking **24**(3), 1841–1852 (Jun 2016). https://doi.org/10.1109/TNET.2015.2438860, http://ieeexplore.ieee.org/document/7128754/

[2] Issa, I., Kamath, S., Wagner, A.B.: Maximal leakage minimization for the Shannon cipher system. In: 2016 IEEE International Symposium on Information Theory (ISIT). pp. 520–524 (Jul 2016). https://doi.org/10.1109/ISIT.2016.7541353

[3] Issa, I., Kamath, S., Wagner, A.B.: An operational measure of information leakage. In: 2016 Annual Conference on Information Science and Systems (CISS). pp. 234–239 (Mar 2016). https://doi.org/10.1109/CISS.2016.7460507

[4] Kocher, P.C.: Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems. In: Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology. pp. 104–113. CRYPTO '96, Springer-Verlag, London, UK, UK (1996), http://dl.acm.org/citation.cfm?id=646761.706156

[5] Liao, J., Kosut, O., Sankar, L., Calmon, F.P.: Privacy Under Hard Distortion Constraints. In: 2018 IEEE Information Theory Workshop (ITW). pp. 1–5 (Nov 2018). https://doi.org/10.1109/ITW.2018.8613385

[6] Liao, J., Sankar, L., Calmon, F.P., Tan, V.Y.F.: Hypothesis testing under maximal leakage privacy constraints. In: 2017 IEEE International Symposium on Information Theory (ISIT). pp. 779–783 (Jun 2017). https://doi.org/10.1109/ISIT.2017.8006634

[7] Nemhauser, G.L., Wolsey, L.A., Fisher, M.L.: An analysis of approximations for maximizing submodular set functions—i. Mathematical Programming **14**(1), 265–294 (Dec 1978). https://doi.org/10.1007/BF01588971, https://doi.org/10.1007/BF01588971

[8] Rockafellar, R.T.: Convex analysis. Princeton Mathematical Series, Princeton University Press, Princeton, N. J. (1970)

[9] Schrijver, A.: Combinatorial Optimization - Polyhedra and Efficiency. Springer (2003)

[10] Wright, C.V., Ballard, L., Coull, S.E., Monrose, F., Masson, G.M.: Spot Me if You Can: Uncovering Spoken Phrases in Encrypted VoIP Conversations. In: 2008 IEEE Symposium on Security and Privacy (sp 2008). pp. 35–49 (May 2008). https://doi.org/10.1109/SP.2008.21

[11] Wu, B., Wagner, A.B., Suh, G.E.: A case for maximal leakage as a side channel leakage metric (2020), https://arxiv.org/abs/2004.08035

[12] Yan, L., Guo, Y., Chen, X., Mei, H.: A study on power side channels on mobile devices. pp. 30–38. ACM Press (2015). https://doi.org/10.1145/2875913.2875934, http://dl.acm.org/citation.cfm?doid=2875913.2875934

[13] Zhang, D., Askarov, A., Myers, A.C.: Predictive mitigation of timing channels in interactive systems. In: Proceedings of the 18th ACM conference on Computer and communications security. pp. 563–574. ACM (2011), http://dl.acm.org/citation.cfm?id=2046772