2019 IEEE 32nd Computer Security Foundations Symposium (CSF)

Quantifying information flow in interactive systems

David Mestel
University of Luxembourg
david.mestel@uni.lu

***Abstract*—We consider the problem of quantifying information flow in interactive systems, modelled as finite-state transducers in the style of Goguen and Meseguer. Our main result is that if the system is deterministic then the information flow is either logarithmic or linear, and there is a polynomial-time algorithm to distinguish the two cases and compute the rate of logarithmic flow. To achieve this we first extend the theory of information leakage through channels to the case of interactive systems, and establish a number of results which greatly simplify computation. We then show that for deterministic systems the information flow corresponds to the growth rate of antichains inside a certain regular language, a property called the width of the language. In a companion work we have shown that there is a dichotomy between polynomial and exponential antichain growth, and a polynomial time algorithm to distinguish the two cases and to compute the order of polynomial growth. We observe that these two cases correspond to logarithmic and linear information flow respectively. Finally, we formulate several attractive open problems, covering the cases of probabilistic systems, systems with more than two users and nondeterministic systems where the nondeterminism is assumed to be innocent rather than demonic.**

***Index Terms*—Quantified information flow, automata theory**

I. Introduction

The notion of 'noninterference' was introduced by Goguen and Meseguer in [1]. It has long been recognised, however, that this condition—that no information can reach Bob about the actions of Alice—may in some circumstances be too strong. The field of quantitative information flow therefore aims to compute the *amount* of information that can reach Bob about Alice's actions.

The contributions of this work are in two main parts. In the first part we extend the theory of information flow through channels developed by Smith, Palamidessi and many others to the case of interactive systems. In addition to basic definitions, we establish a number of results which greatly simplify computation. In particular, we show that it suffices to consider probability distributions over deterministic strategies for the two parties and that one of them may be assumed to adopt a pure deterministic strategy. We also show that if the system itself is deterministic then there is a possiblistic characterisation of the information flow which avoids quantifying over probability distributions altogether; this will be essential for the work of the second part.

In the second part we study determinstic interactive systems modelled as finite-state transducers in the style of Goguen and Meseguer. We define the information-flow capacity of such systems, before addressing the formidable technical problem of computing it. The key idea is to show that this can be reduced to a certain combinatorial problem on partially ordered sets. This problem is solved in a companion work [2], with the consequence that we are able to show (Theorem 30) that for such systems there is a dichotomy between logarithmic and linear information flow, and a polynomial-time algorithm to distinguish the two cases. These two cases are naturally interpreted as 'safe' and 'dangerous' respectively, so we have shown that it is possible to distinguish genuinely dangerous information flow. We thereby accomplish a goal proposed by Ryan, McLean, Millen and Gligor at CSFW'01 in [3].

Overview

In Section II we first recall some relevant theory on the information-flow capacity of channels, and improve a result of Alvim, Chatzikokolakis, McIver, Morgan, Palamidessi and Smith giving an upper bound on the 'Dalenius leakage' of a channel to an exact formula (Theorem 1). We then consider interactive channels, where both parties may be required to make choices. We define leakage and information-flow capacity in this setting, and show that Bob's strategy may be assumed to be deterministic (Corollary 3). We show (Theorem 4) that in the case of deterministic channels we may take a possbilistic view of Alice's actions, which we will find simplifies calculation considerably. Finally we show (Theorem 9) that for systems which may involve multiple rounds of interaction it suffices to consider probability distributions over deterministic, rather than probabilistic, strategies.

In Section III we model deterministic interactive systems as finite-state transducers, and define their information-flow capacity. We then show how to reduce the problem of computing this to a problem involving only nondeterministic finite automata, and then to the combinatorial problem of computing the 'width' of the languages generated by the relevant automata. We observe that this problem is solved in [2], and consequently conclude (Theorem 30) that there is a dichotomy between logarithmic and linear information flow, and a polynomial-time algorithm to distinguish the two cases. The structure of the sequence of reductions leading to this theorem is summarised in Figure 3, and we illustrate the theory by applying it to a simple scheduler.

In Section IV we discuss some generalisations of the systems studied in main part of this work: namely nondeterministic systems, systems with more than two agents (which we observe encompasses the case of nondeterministic systems), and probabilistic systems. For the latter two we define the information-flow capacity and formulate the open problems

The author is supported by FNR under grant number 11689058 (Q-CoDe).

DOI 10.1109/CSF.2019.00035

of computing it. Finally in Section V we discuss related work and in Section VI we conclude.

II. Information-theoretic preliminaries

A. Leakage through channels

We consider first the case of leakage through a channel from a space $\mathcal{X}$ of inputs to a space $\mathcal{Y}$ of outputs, corresponding to a situation in which the attacker is purely passive: Alice selects an input according to a known prior distribution p_X and Bob (the attacker) receives an output according to the conditional distribution $p_{Y|X}$, which specifies the channel. How much information should we say that Bob has received?

The first work on quantified information flow adopted the classical information-theoretic notion of *mutual information* introduced by Shannon in the 1940s [4]. However, Smith observed in [5] the problems with this consensus definition. The essential problem is that mutual information represents in some sense the average number of bits of information leaked by the system. This is appropriate for the noisy coding theorem, where we are interested in the limit of many uses of the channel, but not for the case of information leakage where we assume that the adversary receives only one output (or a small number of outputs).

This means that, in the example used by Smith, a system which leaks the whole secret 1/8 of the time is seen as largely secure (because $H(X|Y) = \frac{7}{8}H(X)$), although it allows (for instance) a cryptographic key to be guessed 1/8 of the time. Smith addresses this by adopting the *min-entropy leakage*, defined[1] as the expected value of the increase in the probability of guessing the input upon observing the output y:

$$\mathcal{L}_\infty(X, Y) = \log \mathbb{E}_{y \sim Y} \frac{\sup_{x \in \mathcal{X}} p_{X|Y}(x|y)}{\sup_{x \in \mathcal{X}} p_X(x)}.$$

Given a channel $\mathcal{C}$ specified by a matrix of conditional probabilities $p_{Y|X}$, we may be interested in its *capacity*, which is the maximum value of the leakage over all possible priors p_X:

$$\mathcal{L}_\infty(\mathcal{C}) = \sup_{(X,Y)\sim\mathcal{C}} \mathcal{L}_\infty(X, Y),$$

where the notation $(X, Y) \sim \mathcal{C}$ means that X and Y are random variables compatible with $\mathcal{C}$; that is, that the conditional probabilities $p_{Y|X}$ (where defined) correspond to the matrix defining $\mathcal{C}$.

In [6], Alvim, Chatzikolakis, Palamidessi and Smith generalise this definition to the notion of *g-leakage*, in which Bob makes a guess drawn from a set $\mathcal{W}$, and receives a payoff according to the function $g : \mathcal{W} \times \mathcal{X} \to [0, 1]$. The leakage with respect to g is then

$$\mathcal{L}_g(X, Y) = \log \mathbb{E}_{y \sim Y} \frac{\sup_{w \in \mathcal{W}} \sum_{x \in \mathcal{X}} p_{X|Y}(x|y) g(w, x)}{\sup_{w \in \mathcal{W}} \sum_{x \in \mathcal{X}} p_X(x) g(w, x)}.$$

Once again, we can define the capacity of a channel $\mathcal{C}$:

$$\mathcal{L}_g(\mathcal{C}) = \sup_{(X,Y)\sim\mathcal{C}} \mathcal{L}_g(X, Y).$$

In Theorem 5.1 of [6], the authors prove the so-called 'miracle' theorem, which states that for any channel $\mathcal{C}$ and any gain function g we have that the g-capacity is at most the min-entropy capacity:

$$\mathcal{L}_g(\mathcal{C}) \le \mathcal{L}_\infty(\mathcal{C}).$$

However, it may be the case that the secret which Bob is trying to guess is not Alice's input but some other secret value (a cryptographic key, say) which is related to x in some known but unspecified way. We may be interested in bounding the possible gain for Bob for any possible secret and any (probabilistic) relationship to the choice of x; this is sometimes known as the 'Dalenius leakage', after a desideratum attributed to T. Dalenius by Dwork in [7]. We may therefore define

$$\mathcal{L}_D(X, Y) = \sup_{Z \in \mathcal{D}} \mathcal{L}_\infty(Z, Y),$$

where $\mathcal{D}$ is the collection of random variables Z such that $Z \to X \to Y$ forms a Markov chain (that is, $p_{X,Y,Z}(x, y, z) = p_Z(z)p_{X|Z}(x|z)p_{Y|X}(y|x)$).

In [8], Alvim, Chatzikokolakis, McIver, Morgan, Palamidessi and Smith give an upper bound for the Dalenius leakage: they show in Corollary 23 that for any Markov chain $Z \to X \to Y$ we have that

$$\sup_g \mathcal{L}_g(Z, Y) \le \sup_g \mathcal{L}_g(Y, X),$$

where the suprema are taken over gain functions g. Hence in particular we have that $\mathcal{L}_\infty(Z, Y) \le \sup_g \mathcal{L}_g(Y, X)$. But $\mathcal{L}_g(Y, X) \le \mathcal{L}_g(\mathcal{C})$, where $\mathcal{C}$ is any channel such that $(X, Y) \sim \mathcal{C}$, and by the miracle theorem we have that $\mathcal{L}_g(\mathcal{C}) \le \mathcal{L}_\infty(\mathcal{C})$, and hence we have that

$$\mathcal{L}_D(X, Y) \le \mathcal{L}_\infty(\mathcal{C}).$$

We are able to improve this to a precise formula for the Dalenius leakage between two random variables.

Theorem 1. *Let X, Y be any discrete random variables. Then*

$$\begin{aligned}\mathcal{L}_D(X, Y) &= \log \mathbb{E}_{y \sim Y} \sup_{x \in \mathcal{X}} \frac{p_{Y|X}(y|x)}{p_Y(y)} \\ &= \log \sum_{y \in \mathcal{Y}^+} \sup_{x \in \mathcal{X}} p_{Y|X}(y|x),\end{aligned}$$

where $\mathcal{Y}^+ \subseteq \mathcal{Y}$ is the set of $y \in \mathcal{Y}$ such that $p_Y(y) > 0$.

Proof. We may assume without loss of generality that $p_Y(y) > 0$ for all $y \in Y$ (otherwise redefine $\mathcal{Y}$ to be the set of values on which p_Y is supported).

For the upper bound, we recall that Braun, Chatzikokolakis and Palamidessi observe in Proposition 5.11 of [9] that there is a simple formula for the min-entropy capacity of a channel $\mathcal{C}$ defined by matrix $p_{Y|X}$:

$$\mathcal{L}_\infty(\mathcal{C}) = \log \sum_{y \in \mathcal{Y}} \sup_{x \in \mathcal{X}} p_{Y|X}(y|x). \tag{1}$$

[1] Smith and subsequent authors generally define leakage only for random variables whose images are finite sets. However, their definitions are straightforwardly generalised to arbitrary discrete random variables by replacing max with sup where appropriate. Except where noted, the proofs of all quoted results remain valid after the same modification.

This is proved in [9] for random variables with finite image. For general discrete random variables, the upper bound on $\mathcal{L}_\infty(\mathcal{C})$ is obtained by replacing $\max$ with $\sup$ as appropriate, but the lower bound requires a little more care since it is given by considering the uniform distribution on $\mathcal{X}$. However, the lower bound can be recovered for infinite $\mathcal{X}$ by considering the uniform distribution on the first k elements of $\mathcal{X}$ and taking the limit as $k \to \infty$.

The upper bound on $\mathcal{L}_D(X, Y)$ is immediate from (1) by taking $\mathcal{C}$ to be any channel such that $(X, Y) \sim \mathcal{C}$.

For the lower bound, suppose that $\mathcal{X} = \{x_1, x_2, x_3, \ldots\}$, and define the function $f : [0, 1) \to \mathcal{X}$ by $f(\xi) = x_k$ if

$$\sum_{i=1}^{k-1} p_X(x_i) < \xi \leq \sum_{i=1}^{k} p_X(x_i).$$

Note that for each $x \in \mathcal{X}$ we have that $p_X(x) = \mu(f^{-1}(x))$, where μ is the Borel measure.

For each positive integer n, let $\mathcal{Z}_n = \{0, 1, 2, \ldots, 2^n - 1\}$ and let Z_n be a random variable taking values in $\mathcal{Z}_n$, with

$$p_{X,Z_n}(x, z) = \mu\left(f^{-1}(x) \cap \left[\frac{z}{2^n}, \frac{z+1}{2^n}\right]\right).$$

Note that by the previous observation we have $\sum_{z \in \mathcal{Z}_n} p_{X,Z_n}(x, z) = \mu(f^{-1}(x)) = p_X(x)$ as required. Note also that we have that $p_Z(z) = \sum_{x \in \mathcal{X}} p_{X,Z_n}(x, z) = 2^{-n}$.

Now we have

$$\begin{aligned}\mathcal{L}_\infty(Z_n, Y) &= \log \sum_{y \in \mathcal{Y}} p_Y(y) \frac{\max_z p_{Z_n|Y}(z|y)}{\max_z p_{Z_n}(z)} \\ &= \log \sum_{y \in \mathcal{Y}} p_Y(y) 2^n \max_z p_{Z_n|Y}(z|y).\end{aligned}$$

We claim that

$$\lim_{n \to \infty} p_Y(y) 2^n \max_z p_{Z_n|Y}(z|y) \geq \sup_x p_{Y|X}(y|x) \qquad (2)$$

for all $y \in \mathcal{Y}$.

Indeed, by Bayes' theorem we have

$$\begin{aligned}p_Y(y) p_{Z_n|Y}(z|y) &= p_{Z_n}(z) p_{Y|Z_n}(y|z) \\ &= 2^{-n} \sum_{x \in \mathcal{X}} p_{Y|X}(y|x) p_{X|Z_n}(x|z).\end{aligned}$$

Let $x \in \mathcal{X}$ be arbitrary. For sufficiently large n we have that $p_{X|Z_n}(x|z) = 1$ for some $z \in \mathcal{Z}_n$, and hence for this z we have that $p_Y(y) 2^n p_{Z_n|Y}(z|y) \geq p_{Y|X}(y|x)$, proving the claim. Summing (2) over all $y \in \mathcal{Y}$ and rearranging gives

$$\lim_{n \to \infty} \mathcal{L}_\infty(Z_n, Y) \geq \log \sum_{y \in \mathcal{Y}} \max_x p_{Y|X}(y|x),$$

as required. □

B. Interactive channels

More generally, we will be interested in *interactive channels*, where an input is chosen by both Alice and Bob, and the system then produces an output to Bob. This means that the space $\mathcal{X}$ is of the form $\mathcal{X}_A \times \mathcal{X}_B$, where the spaces $\mathcal{X}_A$ and $\mathcal{X}_B$ are the spaces of inputs for Alice and Bob respectively, and the interactive channel $\mathcal{C}$ is defined by the matrix of conditional probabilites $p_{Y|X_A,X_B}$.

Note that if the system involves a sequence of outputs and actions by Alice and Bob then the 'inputs' x_A and x_B will in fact represent *strategies* for Alice and Bob, determining their actions on the basis of the outputs they have seen so far (in general these may be probabilistic, but we will see in Section II-D that in fact it is sufficient to consider only deterministic strategies).

We will write $((X_A, X_B), Y) \sim \mathcal{C}$ to mean that the random variables X_A, X_B and Y are consistent with the channel $\mathcal{C}$: that is, that X_A and X_B are independent and the matrix $p_{Y|X_A,X_B}$ corresponds with the matrix defining $\mathcal{C}$.

We can once again define the min-entropy leakage as the expected increase in Bob's probability of guessing the value of the input based on having seen the output:

$$\begin{aligned}&\mathcal{L}_\infty((X_A, X_B), Y) \\ &\quad = \log \mathop{\mathbb{E}}_{x_B \sim X_B, y \sim Y} \frac{\sup_{x_A \in \mathcal{X}_A} p_{X_A|X_B,Y}(x_A|x_B, y)}{\sup_{x_A \in \mathcal{X}_A} p_{X_A}(x_A)} \\ &\quad = \log \mathbb{E}_{x_B \sim X_B} 2^{\mathcal{L}_\infty(X_A, Y | X_B = x_B)}.\end{aligned}$$

Again the capacity of the channel is defined as the maximum leakage over all possible priors p_{X_A} and p_{X_B}.

$$\mathcal{L}_\infty(\mathcal{C}) = \sup_{((X_A, X_B), Y) \sim \mathcal{C}} \mathcal{L}_\infty((X_A, X_B), Y).$$

It appears at first glance that calculating $\mathcal{L}_\infty(\mathcal{C})$ may in general be highly intractible: we have to quantify over mixed strategies (that is over probability distributions on strategies) for Alice and Bob. However, it turns out that we may assume without loss of generality that Bob chooses a pure strategy.[2] Indeed, this holds not only for the choices we have made but for all reasonable such choices.

Specifically, we chose a leakage measure, namely $\mathcal{L}_\infty$, and a method of averaging the leakage over different values of x_B, namely taking $\log \mathbb{E}_{x_B} 2^{\mathcal{L}}$. The following proposition shows that we may assume a pure strategy for Bob for any choice of leakage measure, and any method of averaging which is 'reasonable' in the sense that if the distribution of leakage is constant with value x then the value is x, and also that the value of a weighted sum of leakage distributions cannot be more than the maximum value of the distributions making up the sum (this last property is known as 'quasiconvexity').

Proposition 2. *Let $\mathcal{L} : \mathbb{D}(\mathcal{X}_A \times \mathcal{Y}) \to \mathbb{R}$ (the 'leakage function') be any function and let $\phi : \mathbb{D}(\mathbb{R}) \to \mathbb{R}$ (the*

[2] Note that this means a pure strategy *over the set $\mathcal{X}_B$*, which in an interactive system may contain probabilistic strategies (although we will see in Theorem 9 that these may be ignored without loss of generality).

'averaging function') be any function such that if $X \in \mathbb{D}(\mathbb{R})$ is constant x then $\phi(X) = x$ and for any $X_1, X_2, \ldots \in \mathbb{D}(\mathbb{R})$ and any $\rho_1, \rho_2, \ldots$ with $\sum_i \rho_i = 1$ we have

$$\phi\left(\sum_i \rho_i X_i\right) \leq \sup_i \phi(X_i). \tag{3}$$

Let

$$\mathcal{L}_\phi(\mathcal{C}) = \sup_{((X_A,X_B),Y)\sim\mathcal{C}} \phi(\mathcal{L}(X_A, Y|X_B)).$$

Then we have

$$\mathcal{L}_\phi(\mathcal{C}) = \sup_{x_B\in\mathcal{X}_B} \sup_{(X_A,x_B,Y)\sim\mathcal{C}} \mathcal{L}(X_A, Y),$$

where the notation (X_A, x_B, Y) means the distribution with $p_{X_B}(x_B) = 1$, and in the above $\mathbb{D}(\mathcal{X})$ means the space of probability distributions over the set $\mathcal{X}$.

Proof. Suppose that $(X_A, X_B, Y) \sim \mathcal{C}$. We have

$$\phi(\mathcal{L}(X_A, Y|X_B)) = \phi\left(\sum_{x_B} p_{X_B}(x_B)\mathcal{L}(X_A, Y|X_B = x_B)\right).$$

Hence for any $\epsilon > 0$, by (3) there exists some x_B such that

$$\begin{aligned}\phi(\mathcal{L}(X_A, Y|X_B = x_B)) &= \mathcal{L}(X_A, Y|X_B = x_B)\\ &\geq \phi(\mathcal{L}(X_A, Y|X_B)) - \epsilon.\end{aligned}$$

Hence we have

$$\sup_{x_B\in\mathcal{X}_B} \mathcal{L}(X_A, Y|X_B = x_B) = \phi(\mathcal{L}(X_A, Y|X_B)),$$

establishing the result. □

The min-entropy capacity is a special case of this result, with $\mathcal{L} = \mathcal{L}_\infty$ and $\phi(X) = \log \mathbb{E}_{x\sim X} 2^x$.

Corollary 3. *Let $\mathcal{C}$ be an interactive channel. Then we have*

$$\mathcal{L}_\infty(\mathcal{C}) = \sup_{x_B\in\mathcal{X}_B} \mathcal{L}_\infty(\mathcal{C}|X_B = x_B).$$

C. Deterministic channels

For the channels we have considered above, once the inputs from Alice and Bob are fixed we obtain a probability distribution on outputs. However, for some systems it may be that the output is not probabilistic, but is determined by the values of the inputs; we will call such a channel deterministic. More concretely, an interactive channel $\mathcal{C}$ defined by the matrix $p_{Y|X_A,X_B}$ is *deterministic* if for all x_A, x_B, y we have

$$p_{Y|X_A,X_B}(y|x_A, x_B) \in \{0, 1\}.$$

If $\mathcal{C}$ is deterministic then the computation of $\mathcal{L}_\infty(\mathcal{C})$ simplifies considerably, because it turns out that we can take a purely possibilistic view of Alice's actions and avoid any quantification over probability distributions.

Theorem 4. *Let $\mathcal{C}$ be a deterministic interactive channel. Then*

$$\begin{aligned}\mathcal{L}_\infty(\mathcal{C}) = \sup_{x_B\in\mathcal{X}_B} \log |\{y \in \mathcal{Y}|\exists x_A \in \mathcal{X}_A :&\\ p_{Y|X_A,X_B}(y|x_A, x_B) = 1\}\big|.&\end{aligned}$$

Proof. By Corollary 3, it suffices to prove that

$$\begin{aligned}\mathcal{L}_\infty(\mathcal{C}|X_B = x_B) = \log |\{y \in \mathcal{Y}|\exists x_A \in \mathcal{X}_A :&\\ p_{Y|X_A,X_B}(y|x_A, x_B) = 1\}\big|.&\end{aligned}$$

By the formula for $\mathcal{L}_\infty(\mathcal{C})$ from [9] (recalled as (1) in the proof of Theorem 1) we have

$$\begin{aligned}\mathcal{L}_\infty(\mathcal{C}|X_B = x_B) &= \log \sum_{y\in\mathcal{Y}} \max_{x_A\in\mathcal{X}_A} p_{Y|X_A,X_B}(y|x_A, x_B)\\ &= \log \big|\{y \in \mathcal{Y}|\exists x_A \in \mathcal{X}_A : p_{Y|X_A,X_B}(y|x_A, x_B) = 1\}\big|,\end{aligned}$$

since $\mathcal{C}$ is deterministic and so $p_{Y|X_A,X_B}(y|x_A, x_B) \in \{0, 1\}$. □

Theorem 4 essentially says that it suffices to count the maximum number of outputs that can be seen by Bob, consistently with his choice of strategy. The corresponding result for non-interactive channels is Theorem 1 of [5].

D. Probabilistic vs deterministic strategies

We observed in Section II-B that the 'channel' paradigm is able to model systems involving many rounds of interaction, because we can take Alice and Bob's inputs to be strategies, determining the actions they will take at each step of the interaction. At each step, Alice (respectively Bob) will have observed a trace of the interaction thus far drawn from a set T, and must select an action drawn from a set Σ. To specify a randomised strategy for Alice or Bob, we must therefore specify for each $t \in T$ a probability distribution over Σ, so the set of strategies is the set of maps $T \to \mathbb{D}\Sigma$.

In this section we will show that in fact it suffices to consider only deterministic strategies for Alice and Bob. The intuition behind this is fairly straightforward: given a probabilistic strategy, we could imagine that any necessary coins are tossed before the execution begins, which gives a probability distribution over deterministic strategies. This changes nothing except that it allows Bob to see how the random choices made by his strategy were resolved, but this only gives him more information and so does not affect the information flow capacity. To avoid technical measurability issues we will assume that the sets T and Σ are finite.

Definition 5. *Let T be a finite set of* traces *and Σ a finite set of* actions. *A* strategy *over T and Σ is a function $f : T \to \mathbb{D}(\Sigma)$. The set of strategies over T and Σ is denoted $\mathcal{S}_{T,\Sigma}$.*

A strategy $f \in \mathcal{S}_{T,\Sigma}$ is determinsitic *if we have*

$$f(t)(x) \in \{0, 1\}$$

for all $t \in T$ and $x \in \Sigma$. We write $\mathcal{D}_{T,\Sigma} \subset \mathcal{S}_{T,\Sigma}$ for the set of deterministic strategies over T and Σ.

In the execution itself, these strategies will be executed and particular actions chosen. The output $y \in \mathcal{Y}$ displayed to Bob is then a function (which may be probabilistic) of the choices that were made; the system is defined by this function, which is a map from pairs of functions $T \to \Sigma$ (the choices made by Alice and Bob respectively) to distributions over $\mathcal{Y}$.

Note that it may be that in some executions not all traces are actually presented to Alice and Bob for decision; this can be represented by the choices made in response to those traces being ignored, so no generality is lost by considering total functions $T \to \Sigma$ (similarly the trace-sets relevant to Alice and Bob may be distinct, but this can be represented by ignoring the choices made by Alice on Bob's traces and vice versa).

We write Σ^T for the set of functions $T \to \Sigma$; the probability that a particular function is realised by a particular strategy can be computed by multiplying the probabilities for each decision (note that nothing is lost by assuming independence: if Alice and Bob are supposed to know about previous choices they have made then this can be encoded in the traces).

Definition 6. *Let $f \in \mathcal{S}_{T,\Sigma}$ be any strategy and $g \in \Sigma^T$. The probability that f* realises *g, written $f(g)$, is given by*

$$f(g) = \prod_{t \in T} f(t)(g(t)).$$

Definition 7. *Let $\phi : \Sigma^T \times \Sigma^T \to \mathbb{D}\mathcal{Y}$ be any map, and let $\mathcal{X}_A$ and $\mathcal{X}_B$ be any subsets of $\mathcal{S}_{T,\Sigma}$. The* interactive channel *determined by $\phi, \mathcal{X}_A$ and $\mathcal{X}_B$, denoted $\mathcal{C}_{\phi,\mathcal{X}_A,\mathcal{X}_B}$, is determined by the matrix of conditional probabilities*

$$p_{Y|X_A,X_B}(y|f_A, f_B) = \sum_{g_A, g_B \in \Sigma^T} f_A(g_A) f_B(g_B) \phi(g_A, g_B)(y).$$

We observe that if the function ϕ defining the system is deterministic, and if Alice and Bob use only deterministic strategies, then the channel produced is a deterministic interactive channel in the sense of the previous section, such that Theorem 4 applies to it.

Proposition 8. *Suppose that $\phi(g, g')(y) \in \{0, 1\}$ for every $g, g' \in \Sigma^T$ and $y \in \mathcal{Y}$. Then $\mathcal{C}_{\phi,\mathcal{D}_{T,\Sigma},\mathcal{D}_{T,\Sigma}}$ is a deterministic interactive channel.*

Proof. If $f_A, f_B \in \mathcal{D}_{T,\Sigma}$ then $f_A(g), f_B(g) \in \{0, 1\}$ for all $g \in \Sigma^T$. Hence if $\phi(g_A, g_B, y) \in \{0, 1\}$ for all g_A, g_B, y then we have $p_{Y|X_A,X_B}(y|f_A, f_B) \in \{0, 1\}$ for all f_A, f_B, y, as required. □

The main theorem of this section is that in fact it suffices to conisder only deterministic strategies for Alice and Bob.

Theorem 9. *Let Σ and T be any finite sets, $\mathcal{Y}$ any set and $\phi : \Sigma^T \times \Sigma^T \to \mathbb{D}\mathcal{Y}$ be any map. Then we have*

$$\mathcal{L}_\infty\left(\mathcal{C}_{\phi,\mathcal{S}_{T,\Sigma},\mathcal{S}_{T,\Sigma}}\right) = \mathcal{L}_\infty\left(\mathcal{C}_{\phi,\mathcal{D}_{T,\Sigma},\mathcal{D}_{T,\Sigma}}\right).$$

Proof. The lower bound is immediate: since whenever $((X_A, X_B), Y) \sim \mathcal{C}_{\phi,\mathcal{D}_{T,\Sigma},\mathcal{D}_{T,\Sigma}}$ then also $((X_A, X_B), Y) \sim \mathcal{C}_{\phi,\mathcal{S}_{T,\Sigma},\mathcal{S}_{T,\Sigma}}$, we must have (writing $\mathcal{C}_\mathcal{S}$ and $\mathcal{C}_\mathcal{D}$ resectively for the two channels in the statement of the theorem)

$$\begin{aligned} \mathcal{L}_\infty(\mathcal{C}_\mathcal{S}) &= \sup_{((X_A,X_B),Y)\sim\mathcal{C}_\mathcal{S}} \mathcal{L}_\infty((X_A, X_B), Y) \\ &\geq \sup_{((X_A,X_B),Y)\sim\mathcal{C}_\mathcal{D}} \mathcal{L}_\infty((X_A, X_B), Y) \\ &= \mathcal{L}_\infty(\mathcal{C}_\mathcal{D}). \end{aligned}$$

For the upper bound, let X_A and X_B be any independent $\mathcal{S}_{T,\Sigma}$-valued random variables. We will first show that without loss of generality we may assume that X_B is supported only on $\mathcal{D}_{T,\Sigma}$. By Corollary 3 it suffices to show this where X_B is a point distribution,[3] so say that X_B takes the value $f_B \in \mathcal{S}_{T,\Sigma}$.

Define the random variable X'_B to be supported only on $\mathcal{D}_{T,\Sigma}$, and for $f \in \mathcal{D}_{T,\Sigma}$ let

$$p_{X'_B}(f) = f_B(\widetilde{f}),$$

where $\widetilde{f}$ is the function $T \to \Sigma$ induced by f: that is, $\widetilde{f}(t)$ is the unique element x of Σ such that $f(t)(x) = 1$.

Note that by Definitions 6 and 7 we have that (X_A, X_B) and (X_A, X'_B) induce the same output distribution Y, and so it suffices to prove that for each $y \in \mathcal{Y}$ we have

$$\begin{aligned} &\mathbb{E}_{f'_B \sim X'_B} \sup_{f_A \in \mathcal{S}_{T,\Sigma}} p_{X_A|X'_B,Y}(f_A|f'_B, y) \geq \\ &\qquad\qquad \sup_{f_A \in \mathcal{S}_{T,\Sigma}} p_{X_A|X_B,Y}(f_A|f_B, y). \end{aligned}$$

Now on the one hand we have

$$\begin{aligned} &\mathbb{E}_{f'_B \sim X'_B} \sup_{f_A \in \mathcal{S}_{T,\Sigma}} p_{X_A|X'_B,Y}(f_A|f'_B, y) = \\ &\qquad \sum_{f'_B \in \mathcal{D}_{T,\Sigma}} f_B(\widetilde{f'_B}) \sup_{f_A \in \mathcal{S}_{T,\Sigma}} p_{X_A|X'_B,Y}(f_A|f'_B, y). \end{aligned} \tag{4}$$

On the other hand we have

$$\begin{aligned} &\sup_{f_A \in \mathcal{S}_{T,\Sigma}} p_{X_A|X_B,Y}(f_A|f_B, y) = \\ &\qquad \sup_{f_A \in \mathcal{S}_{T,\Sigma}} \sum_{f'_B \in \mathcal{D}_{T,\Sigma}} f_B(\widetilde{f'_B}) p_{X_A|X'_B,Y}(f_A|f'_B, y). \end{aligned} \tag{5}$$

Plainly (4) $\geq$ (5), establishing the result.

We now show that we may also assume that X_A is supported only on $\mathcal{D}_{T,\Sigma}$, and again by Corollary 3 it suffices to show this where X_B takes only a single value, say $f_B \in \mathcal{D}_{T,\Sigma}$. By the min-entropy capacity formula (1) conditioned on $X_B = f_B$ it suffices to show that for every $y \in \mathcal{Y}$ we have

$$\sup_{f_A \in \mathcal{S}_{T,\Sigma}} p_{Y|X_A,X_B}(y|f_A, f_B) \leq \max_{f_A \in \mathcal{D}_{T,\Sigma}} p_{Y|X_A,X_B}(y|f_A, f_B).$$

But this is straightforward: indeed, for any $f_A \in \mathcal{S}_{T,\Sigma}$ we have

$$\begin{aligned} p_{Y|X_A,X_B}(y|f_A, f_B) &= \sum_{f'_A \in \mathcal{D}_{T,\Sigma}} f_A(\widetilde{f'_A}) \phi(\widetilde{f'_A}, \widetilde{f_B}, y) \\ &\leq \max_{f'_A \in \mathcal{D}_{T,\Sigma}} \phi(\widetilde{f'_A}, \widetilde{f_B}, y) \\ &= \max_{f'_A \in \mathcal{D}_{T,\Sigma}} p_{Y|X_A,X_B}(y|f'_A, f_B, y), \end{aligned}$$

as required. □

[3] Strictly speaking Corollary 3 was proved for discrete distributions, whereas $\mathcal{S}_{T,\Sigma}$ is a continuous subset of $\mathbb{R}^{|T|\cdot|\Sigma|}$. The proof for this case is exactly the same, with sums over $\mathcal{X}_B$ replaced by integrals with respect to the Lebesgue measure.

III. Deterministic interactive systems

A. Finite-state transducers

We will model deterministic interactive systems as deterministic finite-state transducers. Whereas Goguen and Meseguer in [1] modelled such systems as 'state-observed' transducers, we will consider the more general notion of 'action-observed' transducers (see the work of van der Meyden and Zhang in [10] for further discussion of the relationship between noninterference properties in these two models; this model is also essentially equivalent to the notion of 'Input-Output Labelled Transition System' used by Clark and Hunt in the non-quantitative setting in [11]).

Definition 10. *A* deterministic finite-state transducer *(DFST) is a 7-tuple $\mathcal{T} = (Q, q_0, F, \Sigma, \Gamma, \delta, \sigma)$, where Q is a finite set of* states*, $q_0 \in Q$ is the* initial state*, $F \subseteq Q$ is the set of* accepting states*, $\delta : Q \times \Sigma \to Q$ is the* transition function *and $\sigma : Q \times \Sigma \to \Gamma \cup \{\epsilon\}$ is the* output function*.*

A pair $(a_1 a_2 \dots a_k, b_1 b_2 \dots b_l) \in \Sigma^* \times \Gamma^*$ is accepted by $\mathcal{T}$ if there exists a sequence of states $q_1 \dots q_k \in Q^*$ such that $q_k \in F$, for every $0 \leq i < k$ we have $q_{i+1} = \delta(q_i, a_{i+1})$ and $b_1 \dots b_l = \sigma(q_0, a_1)\sigma(q_1, a_2) \dots \sigma(q_{k-1}, a_k)$. We will write $L(\mathcal{T})$ for the subset of $\Sigma^* \times \Gamma^*$ accepted by $\mathcal{T}$; such a set is a deterministic finite-state *transduction*, which we will also abbreviate by DFST.

This definition is not quite convenient for our purposes, because we assume that the agents are able to observe the passage of time. Hence even at a timestep where the machine does nothing, there should be a record in the trace of the fact that time has passed. We ensure this by requiring that there should be an output at each step, and apply the non-standard term 'synchronised' to describe this property (such a transducer is also sometimes called 'letter-to-letter').

Definition 11. *A DFST $\mathcal{T} = (Q, q_0, F, \Sigma, \Gamma, \delta, \sigma)$ is* synchronised *if $\sigma(Q, \Sigma) \subseteq \Gamma$ (that is, we do not have $\sigma(q, a) = \epsilon$ for any $q \in Q$ and $a \in \Sigma$). In this case we say that $\mathcal{T}$ is a synchronised deterministic finite-state transducer (SDFST).*

Note that this definition almost corresponds with the original definition of a *Mealy machine* ([12]), except that we allow for a set of final states $F \neq Q$. It is clear that if $\mathcal{T}$ is synchronised then $(a_1 \dots a_k, b_1 \dots b_l) \in \Sigma^* \times \Gamma^*$ is accepted by $\mathcal{T}$ only if $l = k$. We shall therefore apply the 'zip' operation and view $\mathcal{T}$ as accepting elements of $(\Sigma \times \Gamma)^*$.

We are interested in SDFSTs of a special kind, representing the fact that the system communicates separately with Alice and Bob. We will consider SDFSTs whose input and output alphabets Σ and Γ are of the form $\Sigma_A \times \Sigma_B$ and $\Gamma_A \times \Gamma_B$ respectively. The pairs (Σ_A, Γ_A) and (Σ_B, Γ_B) represent the input and output alphabets used for communication with Alice and Bob respectively.

A simple example of such a transducer is the system which simply relays messages between the two agents (with $\Sigma_A = \Sigma_B = \{a, b\}$ and $\Gamma_A = \Gamma_B = \{a', b'\}$). This is shown in Figure 1.

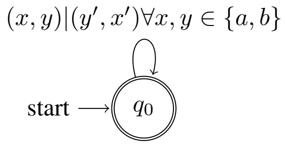

Fig. 1. A relay system.

B. Strategies and information flow

In order to apply the framework of the previous section, we must define the spaces $\mathcal{X}_A, \mathcal{X}_B$ of strategies for Alice and Bob, the space Y of outcomes visible to Bob, and the matrix $p_{Y|X_A,X_B}$ governing which outcomes occur. Since we are considering deterministic specifications, the matrix $p_{Y|X_A,X_B}$ will be 0-1-valued.

Alice and Bob must each decide on an action based on the trace they have seen thus far, so a strategy for Alice is a function

$$x_A : (\Sigma_A \times \Gamma_A)^* \to \Sigma_A,$$

and similarly a strategy for Bob is a function $x_B : (\Sigma_B \times \Gamma_B)^* \to \Sigma_B$.

Recall that by Theorem 9 it suffices to consider deterministic strategies for Alice and Bob: in the language of Section II-D, we have $T = (\Sigma_A \times \Gamma_A)^* \cup (\Sigma_B \times \Gamma_B)^*$ and $\Sigma = \Sigma_A \cup \Sigma_B$. We will have that the function $\phi(g_A, g_B, y)$ ignores the values of g_A on $(\Sigma_B \times \Gamma_B)^*$ and the values of g_B on $(\Sigma_A \times \Gamma_A)^*$, and treats all elements of Σ_B in the image of g_A as equivalent to some fixed $a \in \Sigma_A$ and similarly all elements of Σ_A in the image of g_B as equivalent to some fixed $b \in \Sigma_B$. By Theorem 9 it suffices to consider deterministic strategies for Alice and Bob and so it is more convenient to refer to the sets of deterministic strategies directly as $\mathcal{X}_A$ and $\mathcal{X}_B$, and to $\phi(x_A, x_B)(y)$ directly as the channel matrix $p_{Y|X_A,X_B}(y|x_A, x_B)$.

Given an SDFST $\mathcal{T}$, and strategies x_A and x_B for Alice and Bob respectively, what output or outputs can be shown to Bob? We consider first the case where $F = Q$, postponing for later the issues that arise when $F \subsetneq Q$.

Definition 12. *We will say that a word*

$$w = ((a_1, a_1'), (b_1, b_1')) \dots ((a_k, a_k'), (b_k, b_k')) \\ \in (\Sigma \times \Gamma)^* = ((\Sigma_A \times \Sigma_B) \times (\Gamma_A \times \Gamma_B))^*$$

(so $a_i \in \Sigma_A, a_i' \in \Sigma_B, b_i \in \Gamma_A$ and $b_i' \in \Gamma_B$) is consistent *with SDFST $\mathcal{T}$ and strategies x_A, x_B if*

(i) $w \in L(\mathcal{T})$, and

(ii) for every $1 \leq i \leq k$ we have

$$a_i = x_A((a_1, b_1), \dots, (a_{i-1}, b_{i-1})),$$

and

$$a_i' = x_B((a_1', b_1'), \dots, (a_{i-1}', b_{i-1}')).$$

A word $(a_1', b_1') \dots (a_k', b_k') \in (\Sigma_B \times \Gamma_B)^*$ is consistent with $\mathcal{T}, x_A$ and x_B if there exist $a_1, \dots, a_k \in \Sigma_A$ and $b_1, \dots, b_k \in$

Γ_A *such that* $(((a_1, a'_1), (b_1, b'_1)) \dots ((a_k, a'_k), (b_k, b'_k)))$ *is consistent with* $\mathcal{T}, x_A$ *and* x_B.

We will sometimes refer to limb (ii) of the above Definition as 'being consistent with x_A, x_B'; then being consistent with $\mathcal{T}, x_A, x_B$ means being an element of $L(\mathcal{T})$ and being consistent with x_A, x_B.

Could we choose to have $Y = (\Sigma_B \times \Gamma_B)^*$, and say that $p_{Y|X_A,X_B}(y|x_A, x_B) = 1$ if y is consistent with $\mathcal{T}, x_A$ and x_B?

No, because such a y may not be unique, and so the matrix $p_{Y|X_A,X_B}(y|x_A, x_B)$ would not in general be stochastic. For example, if $\mathcal{T}$ is the identity transduction and x_A and x_B are both constant a, we have that $(a, a)^k$ is consistent with $\mathcal{T}, x_A$ and x_B for all k. But prefixes are the only way this can happen.

Proposition 13. *Let $\mathcal{T}$ be an SDFST as above and let x_A, x_B be strategies for Alice and Bob. Then there exists some $w_0 \in (\Sigma \times \Gamma)^\omega$ such that for any $w \in L(\mathcal{T})$ we have that w is consistent with x_A and x_B if and only if $w \leq w_0$.*

Proof. Define the infinite word

$$w_0 = ((a_1, a'_1), (b_1, b'_1))((a_2, a'_2), (b_2, b'_2)) \dots \in (\Sigma \times \Gamma)^\omega$$

by

$$\begin{aligned} a_i &= x_A((a_1, b_1) \dots (a_{i-1}, b_{i-1})), \\ a'_i &= x_B((a'_1, b'_1) \dots (a'_{i-1}, b'_{i-1})), \text{ and} \\ (b_i, b'_i) &= \sigma(q_{i-1}, (a_i, a'_i)), \end{aligned}$$

where q_0 is the initial state and the sequence $q_0 q_1 \dots$ is defined by $q_i = \delta(q_{i-1}, (a_i, a'_i))$ for $i \geq 1$.

Clearly if $w \leq w_0$ then w satisfies limb (ii) of Definition 12, and so if also $w \in L(\mathcal{T})$ then w is consistent with $\mathcal{T}, x_A$ and x_B.

Conversely suppose that $w \not\leq w_0$. Then we have that $w = w'(a, b)w''$ for some $w' = ((a_1, a'_1), (b_1, b'_1)) \dots ((a_k, a'_k), (b_k, b'_k)) \leq w_0$, some $w'' \in \Sigma \times \Gamma^*$ and some $(a, b) \in \Sigma \times \Gamma$ with $(a, b) \neq ((a_{k+1}, a'_{k+1}), (b_{k+1}, b'_{k+1}))$. But if $a \neq (a_{k+1}, a'_{k+1})$ then without loss of generality we have $\mathrm{fst}(a) \neq a_{k+1} = x_A((a_1, b_1) \dots (a_k, b_k))$ and so w is not consistent with x_A, x_B.

On the other hand if $b \neq (b_{k+1}, b'_{k+1}) = \sigma(q_k, (a_k, a'_k))$ then $w \notin L(\mathcal{T})$. Either way we have that w is not consistent with $\mathcal{T}, x_A$ and x_B. □

The intuition here is that having fixed x_A and x_B, these uniquely determine the actions of Alice and Bob at each step given the outputs they are shown, and $\mathcal{T}$ determines those outputs uniquely based on the actions up to the current time.

Projecting w_0 onto $(\Sigma_B \times \Gamma_B)^\omega$ gives

Corollary 14. *Let $\mathcal{T}, x_A$ and x_B be as above. There exists some $w_0 \in (\Sigma_B \times \Gamma_B)^\omega$ such that if $w \in (\Sigma_B \times \Gamma_B)^*$ is consistent with $\mathcal{T}, x_A$ and x_B then $w \leq w_0$.*

So can we have $Y = (\Sigma_B \times \Gamma_B)^\omega$, and $p_{Y|X_A,X_B}(y|x_A, x_B) = 1$ for $y = w_0$ as in Corollary 14?

One reason why not is that this is not at all realistic: it corresponds to Bob being able to conduct an experiment lasting for an infinite time. Moreover it would allow Bob to acquire an infinite (or at least unbounded) amount of information, and it is not clear how this should be interpreted.

For this reason we will consider Bob's interaction with the system not as a single experiment, but as a *family* of experiments, parametrised by the amount of time allowed; that is, by the length of traces which we consider as outcomes. Assuming for the moment that $F = Q$, we then have that the matrix $p_{Y|X_A,X_B}$ is stochastic.

Proposition 15. *Let $\mathcal{T}$ be an SDFST with $F = Q$, and let $Y = (\Sigma_B \times \Gamma_B)^k$ for some $k \in \mathbb{N}$. Let the matrix $p_{Y|X_A,X_B}$ be defined by $p_{Y|X_A,X_B}(y|x_A, x_B) = 1$ if y is compatible with $\mathcal{T}, x_A$ and x_B, and 0 otherwise. Then $p_{Y|X_A,X_B}$ is stochastic; that is, we have*

$$\sum_{y \in Y} p_{Y|X_A,X_B}(y|x_A, x_B) = 1$$

for all $x_A \in \mathcal{X}_A$ and $x_B \in \mathcal{X}_B$.

Proof. By Corollary 14, we have that for fixed x_A, x_B there is at most one $y \in (\Sigma_B \times \Gamma_B)^k$ which is consistent with $\mathcal{T}, x_A$ and x_B. On the other hand it is clear from the definitions that if $F = Q$ then all prefixes of the infinite word w_0 from Proposition 13 are accepted by $\mathcal{T}$. Hence projecting w_0 onto $(\Sigma_B \times \Gamma_B)^k$ gives a suitable y. □

Truncating at length k also means that strategies x_A, x_B can be viewed as drawn from the spaces of functions $(\Sigma_A \times \Gamma_A)^{<k} \to \Sigma_A$ and $(\Sigma_B \times \Gamma_B)^{<k} \to \Sigma_B$ respectively. This means that the spaces $\mathcal{X}_A$ and $\mathcal{X}_B$ of possible strategies for Alice and Bob are also finite.

We can now apply Theorem 4 to calculate the information flow as the size of the largest possible set of outcomes that can consistently be seen by Bob, and for convenience we will adopt this as a definition.

Definition 16. *Let $\mathcal{T}$ be an SDFST over input and output alphabets $\Sigma_A \times \Sigma_B$, and let $\mathcal{X}_A, \mathcal{X}_B$ be the spaces of functions $(\Sigma_A \times \Gamma_A)^* \to \Sigma_A$ and $(\Sigma_B \times \Gamma_B)^* \to \Sigma_B$ respectively. Define*

$$\begin{aligned} \mathcal{L}_k(\mathcal{T}) = \max_{x_B \in \mathcal{X}_B} \log \big| \{ y \in (\Sigma_B \times \Gamma_B)^k \big| \exists x_A \in \mathcal{X}_A : \\ y \textit{ is consistent with } \mathcal{T}, x_A \textit{ and } x_B \} \big| . \end{aligned}$$

Observe that if $F = Q$ then by Theorem 4 we have that $\mathcal{L}_k(\mathcal{T}) = \mathcal{L}_\infty(\mathcal{C})$, where $\mathcal{C}$ is the interactive channel defined by the matrix of conditional probabilities in the statement of Proposition 15.

What about the case where $F \subsetneq Q$? The treatment of this depends on what we consider to be the meaning of a run ending in a non-accepting state. One interpretation is that it represents a catastrophically bad outcome (say, the intruder being detected) which must be avoided. By Corollary 3 we may assume that Bob is employing a pure (i.e. non-random) strategy, and so Alice can ensure that non-accepting runs are avoided by avoiding particular x_A. This means that Definition 16 is exactly right for this interpretation.

Another possible interpretation is that a run ending in a non-accepting state produces some kind of 'error' output, where all errors are indistinguishable. This essentially increases the number of possible observations by Bob by either 1 or 0, depending on whether or not the extremal x_B allows for non-accepting runs. This means that the amount of information is either $\mathcal{L}_k(\mathcal{T})$ or $\log(1 + 2^{\mathcal{L}_k(\mathcal{T})})$, which we consider to be a trivial difference.

A third possiblity of course is that we reject the very notion of a non-accepting run, and consider only SDFSTs with $F = Q$. Note that many kinds of behaviour which may involve the system going into an 'error' state and producing only a fixed 'dummy' output symbol can straightforwardly be modelled as an SDFST with $F = Q$.

Which of these three options the reader considers most satisfactory is, to some extent, a matter of personal taste. However, since as noted above all are modelled adequately by Definition 16, that is what we shall adopt as the basic definition for the remainder of this analysis.

Definition 16 is in some sense an intensional definition, in the sense that it involves directly considering all possible strategies for Alice and Bob. It will be helpful to have a more extensional version. Definition 16 can be recast as

$$\mathcal{L}_k(\mathcal{T}) = \max_{X \in \mathcal{F}} \log |X|,$$

where $\mathcal{F} \subseteq \mathcal{P}((\Sigma_B \times \Gamma_B)^k)$ is the family of sets X such that there exists some $x_B \in \mathcal{X}_B$ such that

$$X = \left\{ y \in (\Sigma_B \times \Gamma_B)^k \middle| \exists x_A \in \mathcal{X}_A : \right. \\ \left. y \text{ is consistent with } \mathcal{T}, x_A \text{ and } x_B \right\}.$$

So having an extensional characterisation of $\mathcal{L}_k(\mathcal{T})$ amounts to having a condition for a set X to be a member of $\mathcal{F}$.

Theorem 17. *Let $\mathcal{T}, \mathcal{X}_A$ and $\mathcal{X}_B$ be as above. Let $\mathcal{F} \subseteq \mathcal{P}((\Sigma_B \times \Gamma_B)^*)$ be defined by $Y \in \mathcal{F}$ if and only if there exists some $x_B \in \mathcal{X}_B$ such that*

$$Y = \{ y \in (\Sigma_B \times \Gamma_B)^* | \exists x_A \in \mathcal{X}_A : \\ y \textit{ is consistent with } \mathcal{T}, x_A \textit{ and } x_B \}.$$

Let $X \subseteq (\Sigma_B \times \Gamma_B)^*$ be arbitrary. Then $X \subseteq X'$ for some $X' \in \mathcal{F}$ if and only if

(i) $X \subseteq L(\mathcal{T})|_{(\Sigma_B \times \Gamma_B)^*}$, and

(ii) X does not contain two elements which first differ by an element of Σ_B. That is, we do not have $w_1, w_2 \in X$ such that $w_1 = w(a, b)w'$ and $w_2 = w(a', b')w''$ with $w, w', w'' \in (\Sigma_B \times \Gamma)^*, a, a' \in \Sigma_B$ and $b, b' \in \Gamma_B$ with $a \neq a'$,

where the notation $L(\mathcal{T})|_{(\Sigma_B \times \Gamma_B)^*}$ means the projection of $L(\mathcal{T}) \subseteq ((\Sigma_A \times \Sigma_B) \times (\Gamma_A \times \Gamma_B))^*$ onto the set $(\Sigma_B \times \Gamma_B)^*$.

Proof. The 'only if' direction is straightforward. Part (i) is immediate from the definitions, and for part (ii) we must have $a = x_B(w) = a'$ (for the relevant x_B).

For the 'if' direction, suppose that X satisfies the two conditions in the statement of the theorem. Define the partial function $x : (\Sigma_B \times \Gamma_B)^* \rightharpoonup \Sigma_B$ by $x(w') = a$ whenever $w'(a, b) \leq w$ for some $w \in X$ and some $b \in \Gamma_B$. This is well-defined by condition (ii). Define $x_B : (\Sigma_B \times \Gamma_B)^* \to \Sigma_B$ to be x, extended arbitrarily where x is undefined. We claim that

$$X \subseteq Y = \{ y \in (\Sigma_B \times \Gamma_B)^* | \exists x_A \in \mathcal{X}_A : \\ y \text{ is consistent with } \mathcal{T}, x_A \text{ and } x_B \}.$$

Indeed, let $w \in X$ be arbitrary. Plainly w is consistent with x_B. Since $w \in L(\mathcal{T})|_{(\Sigma_B \times \Gamma_B)^*}$, there exists some $w' \in L(\mathcal{T})$ such that $w'|_{(\Sigma_B \times \Gamma_B)^*} = w$. Define the partial function $x' : (\Sigma_A \times \Gamma_A)^* \rightharpoonup \Sigma_A$ by $x'(w'') = a$ whenever $w''(a, b) \leq w'$ for some $b \in \Gamma_A$. Let $x_A : (\Sigma_A \times \Gamma_A)^* \to \Sigma_A$ be an arbitrary total extension of x'. Then plainly w' is consistent with x_A, and is also consistent with x_B since w was. Hence w is consistent with $\mathcal{T}, x_A$ and x_B, as required. □

Truncating to length k, and observing that

$$\max_{X \in \mathcal{F}} \log |X| = \max_{X \subseteq X' \in \mathcal{F}} \log |X|$$

gives

Corollary 18. *Let $\mathcal{T}, \mathcal{X}_A$ and $\mathcal{X}_B$ be as above. Then we have*

$$\mathcal{L}_k(\mathcal{T}) = \max_{X \in \mathcal{F}'_k} \log |X|,$$

where $\mathcal{F}'_k \subseteq \mathcal{P}\left(L(\mathcal{T})_{=k}|_{(\Sigma_B \times \Gamma_B)^k} \right)$ is the collection of sets which do not contain two words which first differ by an element of Σ_B (and this has the same meaning as in part (ii) of Theorem 17).

C. Reduction to automata

In this section, we show how to reduce the problem of computing $\mathcal{L}_k(\mathcal{T})$ from a problem about transducers to a problem which mentions only automata. The first step is to produce an automaton whose language is in correspondence with Bob's interface with $\mathcal{T}$.

Definition 19. *Let $\mathcal{T} = (Q, q_0, F, \Sigma_A \times \Sigma_B, \Gamma_A \times \Gamma_B, \delta, \sigma)$ be an SDFST. Define the nondeterministic finite automaton $\mathcal{A}_{\mathcal{T}} = (Q \cup (Q \times \Gamma_B), q_0, F, \Sigma_B \cup \Gamma_B, \Delta)$, where*

$$\Delta(q, a') = \{ (\delta(q, (a, a')), \mathrm{snd}(\sigma(q, (a, a')))) | a \in \Sigma_A \}$$

for all $q \in Q$ and $a' \in \Sigma_B$, $\Delta(q, b') = \emptyset$ for all $b' \in \Gamma_B$, and

$$\Delta((q, b'), x) = \begin{cases} \{q\} & \textit{if } x = b' \\ \emptyset & \textit{otherwise} \end{cases}$$

for all $(q, b') \in Q \times \Gamma_B$ and $x \in \Sigma_B \cup \Gamma_B$.

Informally, we introduce an auxiliary state for each pair $(q, b') \in Q \times \Gamma_B$ to represent the behaviour 'emit the event b' and then go into state q'. For states $q, q' \in Q$ and events $a' \in \Sigma_B, b' \in \Gamma_B$ we have a transition from q to (q', b') if and only there exist some $a \in \Sigma_A$ and $b \in \Gamma_A$ such that

$\delta(q,(a,a')) = q'$ and $\sigma(q,(a,a')) = (b,b')$. In the language of Communicating Sequential Procceses, this corresponds to treating Alice's behavious as nondeterministic and hiding all of her events: that is, the familiar *lazy abstraction* formulation of noninterference [13].

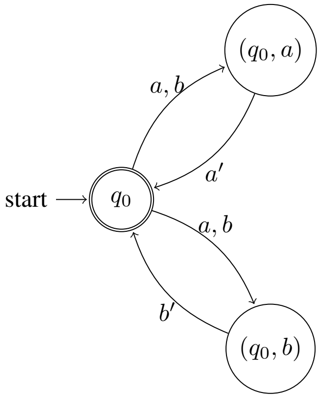

Fig. 2. Automaton corresponding to the relay system transducer shown in Figure 1.

The following lemma is immediate from the definitions, and expresses the fact that the words accepted by $\mathcal{A}_\mathcal{T}$ are in precise correspondence with the words accepted by $\mathcal{T}$, projected onto $\Sigma_B \times \Gamma_B$.

Lemma 20. *Let $f : (\Sigma_B \times \Gamma_B)^* \to (\Sigma_B \cup \Gamma_B)^*$ be the* flattening *operation defined by $f((a_1, b_1)\ldots(a_k, b_k)) = a_1 b_1 \ldots a_k b_k$. Then we have*

$$L(\mathcal{A}_\mathcal{T}) = f\left(L(\mathcal{T})|_{(\Sigma_B \times \Gamma_B)^*}\right).$$

Note that since elements of Σ_B appear at odd-numbered positions in traces of $\mathcal{A}_\mathcal{T}$ and elements of Γ_B appear at even-numbered positions, we may assume without loss of generality that Σ_B and Γ_B are disjoint. Then combining Lemma 20 with Corollary 18 gives

Theorem 21. *Let $\mathcal{T}$ be an SDFST as above such that Σ_B and Γ_B are disjoint. Then*

$$\mathcal{L}_k(\mathcal{T}) = \max_{X \in \mathcal{F}_k} \log |X|,$$

where $\mathcal{F}_k \subseteq \mathcal{P}\left(L\left(\mathcal{A}_\mathcal{T}\right)_{=2k}\right)$ is the collections of sets which do not contain two words which first differ by an element of Σ_B; that is, for $X \in \mathcal{F}_k$ we do not have $w_1, w_2 \in X$ with $w_1 = waw', w_2 = wa'w''$, with $w, w', w'' \in (\Sigma_B \cup \Gamma_B)^*$ and $a \neq a' \in \Sigma_B$.

Note that an alternative notation for this theorem (and, *mutatis mutandis*, Corollary 18) would be to define a single family $\mathcal{F} \subseteq \mathcal{P}\left((\Sigma_B \cup \Gamma_B)^*\right)$ consisting of the sets which do not contain words first differing on an element of Σ_B, and then say that

$$\mathcal{L}_k(\mathcal{T}) = \max_{X \in \mathcal{F}} \log |X \cap L(\mathcal{A}_\mathcal{T})_{=2k}|.$$

We have therefore reduced computing the information flow permitted by a deterministic interactive system to an instance of a more general problem over finite automata, which we call the Σ-*deterministic subset growth* problem.

Definition 22. *Let Σ, Γ be disjoint finite sets. A set $X \subseteq (\Sigma \cup \Gamma)^*$ is* Σ-deterministic *if it does not contain two words which first differ by an element of Σ; that is, we do not have $w_1, w_2 \in X$ with $w_1 = waw', w_2 = wa'w''$, with $w, w', w'' \in (\Sigma \cup \Gamma)^*$ and $a \neq a' \in \Sigma$.*

For a nondeterministic finite automaton $\mathcal{A}$ over alphabet $\Sigma \cup \Gamma$, define

$$D_k(\mathcal{A}) = \max_{X \in \mathcal{F}_k} |X|,$$

where $\mathcal{F}_k$ consists of the Σ-deterministic subsets of $L(\mathcal{A})_{=k}$.

Problem 23 (Σ-deterministic subset growth)**.** *Given a nondeterministic finite automaton $\mathcal{A}$ over $\Sigma \cup \Gamma$, determine the growth rate of $D_k(\mathcal{A})$.*

Of course, the statement of this problem is somewhat informal, in that the meaning of 'determine the growth rate' is not precisely specified. This is in some sense inevitable, considering that $D_k(\mathcal{A})$ is an infinite collection of values, so many types of results are possible. Below we will obtain results on the asymptotic growth of $D_k(\mathcal{A})$ as $k \to \infty$.

D. Antichains

In this section we will see that Problem 23 can be further reduced, to that of computing the 'width' of $L(\mathcal{A})$.

Definition 24. *Let X be a set, and let $\leq$ be a partial order on X. Then the* lexicographic *order induced by $\leq$ on X^*, denoted $\preceq$, is defined by*

$$\begin{aligned} &\forall w \in X^* : \epsilon \preceq w \text{ (and } w \not\preceq \epsilon \text{ if } w \neq \epsilon\text{), and} \\ &\forall x, y \in X, w, w' \in X^* : xw \preceq yw' \text{ if and only if either } x < y \\ &\qquad\qquad\qquad \text{or } x = y \text{ and } w \preceq w'. \end{aligned}$$

Observe that $\preceq$ defines a partial order. Indeed, suppose that $w_1, w_2 \in X^*$ are of minimum total length such that $w_1 \preceq w_2, w_2 \preceq w_1$ but $w_1 \neq w_2$. Trivially if $w_1 = \epsilon$ then also $w_2 = \epsilon = w_1$ (and vice versa). Otherwise we have $w_1 = xw, w_2 = yw'$, and either $x < y$ or $x = y$ and $w \preceq w'$, and on the other hand either $y < x$ or $y = x$ and $w' \preceq w$. Hence we have $y = x$ and both $w \preceq w'$ and $w' \preceq w$, so by induction $w = w'$. Hence $w_1 = w_2$, a contradiction, so indeed $\preceq$ is antisymmetric.

Similarly suppose that $w_1, w_2, w_3 \in X^*$ are of minimum total length such that $w_1 \preceq w_2$ and $w_2 \preceq w_3$ but $w_1 \not\preceq w_3$. Since $w_1 \not\preceq w_3$ we have $w_1 \neq \epsilon$, and plainly $w_1 \neq w_2$ and $w_2 \neq w_3$ and so $w_2, w_3 \neq \epsilon$. Write $w_1 = xw, w_2 = yw'$ and $w_3 = zw''$. If $x < y$ then (since $y \leq z$) we have $x < z$ and so $w_1 \preceq w_3$. Similarly if $y < z$ then (since $x \leq y$) we have $x < x$ so $w_1 \preceq w_3$. Hence we have $x = y = z$ and $w \preceq w'$ and $w' \preceq w''$. But then by induction we have $w \preceq w''$ and so $w_1 \preceq w_3$, a contradiction. Hence indeed $\preceq$ is transitive and so (since we have also shown it is antisymmetric, and it is trivially reflexive) it is a partial order.

$$\begin{tikzcd}
\sup_{((X_A,X_B),Y)} \mathcal{L}_\infty((X_A,X_B),Y) \arrow[d, "\text{Theorem 4}"'] \\
\sup_{x_B\in\mathcal{X}_B} \log\left|\{y\in\mathcal{Y}|\exists x_A\in\mathcal{X}_A : p_{Y|X_A,X_B}(y|x_A,x_B)=1\}\right| \arrow[d, "\text{Corollary 18}"'] \\
\max\left\{\log|X|\middle| X\subseteq L(\mathcal{T})|_{(\Sigma_B\times\Gamma_B)^k}, \text{ no } w,w'\in X \text{ first differ in } \Sigma_B\right\} \arrow[d, "\text{Theorem 21}"'] \\
\max\{\log|X||X\subseteq L(\mathcal{A}_\mathcal{T})_{=2k} \text{ is } \Sigma_B\text{-deterministic}\} \arrow[d, "\text{Theorem 27}"'] \\
\max\{\log|X||X\subseteq L(\mathcal{A}_\mathcal{T})_{=2k} \text{ is an antichain}\}
\end{tikzcd}$$

Fig. 3. The structure of Sections II and III

The study of partially ordered sets is often concerned with *chains* (sets wherein any two elements are comparable) and *antichains* (sets where no two elements are comparable).

Definition 25. *Let X be a partially ordered set, partially orderd by $\leq$. A set $Y \subseteq X$ is a* chain *if for any $x, y \in Y$ we have $x \leq y$ or $y \leq x$. Y is an* antichain *if for any $x, y \in Y$ such that $x \leq y$ we have $x = y$. Let $Y \subseteq X$ be an antichain of maximum size. Then $|Y|$ is the* width *of X, denoted $w(X)$.*

An example of the relevance of the width of a partially ordered set to its structure is given by the celebrated theorem of Robert Dilworth [14].

Theorem 26 (Dilworth, 1950)**.** *Let X be a partially ordered set. Let k be minimal such that $X = Y_1 \cup \ldots \cup Y_k$ with each Y_k a chain. Then $k = w(X)$.*

The relevance of these ideas to Problem 23 is established by the following theorem.

Theorem 27. *Let Σ, Γ be disjoint sets. Define the partial order $\leq$ on $\Sigma \cup \Gamma$ by setting $\leq|_\Sigma$ to be an arbitrary linear order on Σ, and setting $x \not\leq y, y \not\leq x$ for all $x \in \Gamma$ and all $y \in \Sigma \cup \Gamma$ with $y \neq x$.*

Let $X \subseteq (\Sigma \cup \Gamma)^k$ be arbitrary. Then X is Σ-deterministic if and only if it is an antichain with respect to the lexicographic order induced by $\leq$.

Proof. If $w_1, w_2 \in (\Sigma \cup \Gamma)^k$ first differ by an element of Σ, say $w_1 = waw'$, $w_2 = wa'w''$ with $a \neq a' \in \Sigma$. Then without loss of generality $a < a'$, so $w_1 \preceq w_2$. Conversely, if $w_1 \preceq w_2$, then write $w_1 = wxw', w_2 = wyw''$ for some $x \neq y \in \Sigma \cup \Gamma$. But then we must have $x < y$, and hence $x, y \in \Sigma$ and so w_1, w_2 first differ by an element of Σ. □

We have thus reduced Problem 23 to the problem of calculating the growth rate of the width of a regular language, with respect to this partial order, a special case of the following problem.

Problem 28 (Antichain growth for NFA)**.** *Given a nondeterministic finite automaton $\mathcal{A}$ over a finite partially ordered set $(\Sigma, \leq)$, determine the growth rate of $w\left(L(\mathcal{A})_{=k}\right)$, with respect to the lexicographic order.*

The structure of the reductions in the preceding sections is shown in Figure 3.

Problem 28 is solved in [2], the relevant results of which are summarised in Theorem 29 (Theorems 16, 18, 25 and 28 of [2]).

Theorem 29. *Let $\mathcal{A}$ be an NFA over a partially ordered set $(\Sigma, \leq)$. Then we have the following:*

(i) The antichain growth of $L(\mathcal{A})$ is either polynomial or exponential. That is, we have either $w(L(\mathcal{A})_{=n}) = O(n^k)$ for some k or $w(L(\mathcal{A})_{=n}) = \Omega(2^{\epsilon n})$ for some $\epsilon > 0$.

(ii) There is a polynomial-time algorithm to determine whether a given $\mathcal{A}$ has polynomial or exponential antichain growth.

(iii) In the case of polynomial antichain growth, we have that $w(L(\mathcal{A})_{=n}) = \Theta(n^k)$ for some integer k, and there is a polynomial-time algorithm to compute k for a given automaton.

Combining Theorem 29 with the reduction shown in Figure 3 yields the main theorem of this work, that any SDFST has either logarithmic or linear min-entropy capacity, and there is a polynomial-time algorithm to distinguish the two cases (and determine the constant for logarithmic capacity).

Theorem 30. *Let $\mathcal{T} = (Q, q_0, F, \Sigma_A \times \Sigma_B, \Gamma_A \times \Gamma_B, \delta, \sigma)$ be an SDFST. Then we have the following:*

(i) The min-entropy capacity $\mathcal{L}_n(\mathcal{T})$ is either logarithmic or linear. That is, we have either $\mathcal{L}_n(\mathcal{T}) = O(\log n)$ or $\mathcal{L}_n(\mathcal{T}) = \Theta(n)$.

(ii) There is a polynomial-time algorithm to determine whether a given $\mathcal{T}$ has logarithmic or linear capacity growth.

(iii) In the case of logarithmic capacity, we have that $\mathcal{L}_n(\mathcal{T}) \sim k \log n$ for some integer k, and there is a polynomial-time algorithm to compute k for a given SDFST.

Note in particular that the information flow capacity is bounded if and only if $w(L(\mathcal{A})_{=n})$ has polynomial growth of order 0.

Returning to the relay system shown in Figure 1 at the beginning of this section, it is easy to see that the corresponding automaton shown in Figure 2 has exponential antichain growth, since in particular its language contains the exponential antichain $(aa'+ab')^*$. We conclude that the system allows linear information flow, which is as expected since in n steps Alice can transmit n independent bits to Bob.

We claim that the cases of linear and logarithmic information flow can in some sense be interpreted as 'dangerous' and 'safe' respectively. That linear information flow is dangerous should require no explanation: it offers an adversary an exponential speedup over exhaustive guessing of a secret (for instance a cryptographic key). On the other hand, if the information flow in time n is only proportional to $\log n$, then this offers the adversary at most a polynomial speedup over exhaustive guessing.

Of course it will not be appropriate in every situation to regard logarithmic antichain growth as 'safe', and for instance we may sometimes be more interested in the precise amount of information flow that can occur in a fixed time n. This is given by $w(L(\mathcal{A}_\mathcal{T})_{=n})$, which can be computed by a straightforward dynamic programming algorithm at the cost of determinising $\mathcal{A}_\mathcal{T}$; see p.89 of the author's PhD thesis [15] for details. Whether there is an algorithm which is polynomial in n and the size of $\mathcal{A}_\mathcal{T}$ (as an NFA) is an open problem.

E. Example: a simple scheduler

We now illustrate the theory of the preceding two sections by applying it to analyse a simple scheduler. A resource is shared between Alice and Bob, and at each step Alice can transmit a, signifying that she wishes to use the resource, or b, signifying that she does not. She receives back either an a', signifying that she was succesful, or a b', signifying that she was not (if she did not ask to use the resource then she always receives a b'). The interface for Bob is similar but with primed and unprimed alphabets reversed.

Initially, Bob has priority over the use of the system, and for as long as Alice transmits b he retains it. However, as soon as Alice seeks to use the system by transmitting an a she obtains priority and retains it for as long as she uses it continuously. As soon as she transmits a b priority shifts back to Bob, who retains it for the remainder of the execution.

The transducer $\mathcal{T}$ corresponding to this system is depicted in Figure 4 (where missing arguments mean that the input from that user is ignored).

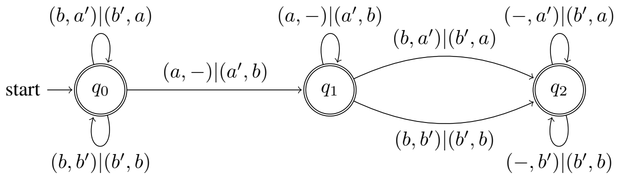

Fig. 4. An interrupt system.

We can now apply Definition 19 to construct the corresponding automaton $\mathcal{A}$, which is shown in Figure 5. By Theorems 21 and 27 we have that $\mathcal{L}_n(\mathcal{T}) = w(L(\mathcal{A})_{=k})$, where $L(\mathcal{A})$ is given the lexicographic order with the primed letters linearly ordered and the unprimed letters incomparable.

By the criteria in Theorems 16 and 28 of [2], this automaton has polynomial antichain growth of order 2, and so the system has logarithmic information flow, with $\mathcal{L}_n(\mathcal{T}) \sim 2\log n$ (see Section 6.5.2 of [15] for a more detailed discussion). Note that this makes intuitive sense: Alice can choose when to start using the resource and when to stop, which she can do in $\binom{n}{2} = \Theta(n^2)$ ways.

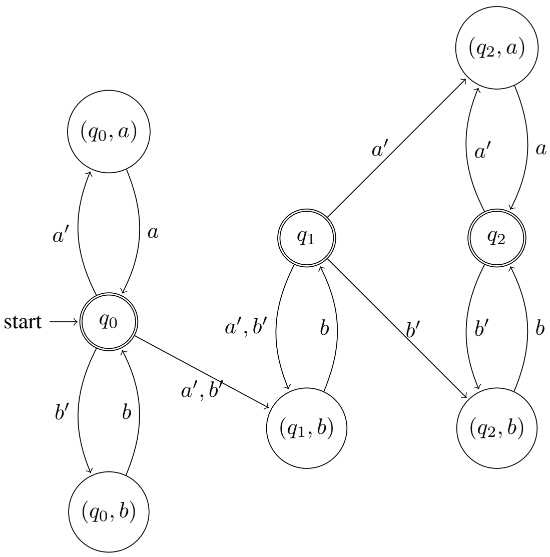

Fig. 5. Automaton corresponding to the interrupt system transducer shown in Figure 4.

IV. Nondeterministic, multi-agent and probabilistic systems

In this section we describe some open problems relating to various generalisations of the deterministic, two-agent systems considered in Section III.

A. Nondeterministic systems

In Section III we considered only deterministic systems. More generally, however, we may be interested in systems which are nondeterministic:

Definition 31. *A* synchronised nondeterministic finite-state transducer *(SNDFST) is a 6-tuple* $\mathcal{T} = (Q, q_0, F, \Sigma, \Gamma, \Delta)$, *where* Q, q_0 *and* F *are as in the definition of DFST, and* $\Delta \subseteq Q \times \Sigma \times Q \times \Gamma$ *is the* transition relation.

Similarly to before we say that $(a_1a_2\ldots a_k, b_1b_2\ldots b_k) \in \Sigma^* \times \Gamma^*$ is accepted by $\mathcal{T}$ if there exists a sequence of states $q_1 \ldots q_k \in Q^*$ such that $q_k \in F$ and for every $0 \leq i < k$ we have $(q_i, a_i, q_{i+1}, b_i) \in \Delta$. As before we will consider systems for which $\Sigma = \Sigma_A \times \Sigma_B$ and $\Gamma = \Gamma_A \times \Gamma_B$, representing the inputs and outputs of Alice and Bob respectively.

The question then arises of how the nondeterminism in the system should be interpreted. One option is to consider it is essentially 'demonic'—that is, under the control of Alice and available to be used to convey information to Bob. This precisely corresponds to Definition 16, which can be adopted wholesale, and a construction similar to that in Definition 19 can be used to produce an NFA $\mathcal{A}_\mathcal{T}$ such that the capacity of

$\mathcal{T}$ is equivalent to the antichain growth of $\mathcal{A}$. We therefore have that Theorem 30 holds also for nondeterministic systems interpreted in this way.

However, the assumption of demonic nondeterminism may in some circumstances be too pessimistic. In particular, it may sometimes be reasonable to assume that the way the nondeterminism is resolved depends only on the previous events, and not on Alice's secret. Equivalently, we may imagine that the resolution of the nondeterminism is controlled by an 'innocent' third party who is isolated from both Alice and Bob (but is able to see their inputs and outputs). We thus have that if we can handle deterministic systems with multiple agents then we will be able to handle nondeterministic systems with this interpretation.

B. Multi-agent systems

We will model multi-agent systems as SDSFTs, as before, but now with

$$\Sigma = \Sigma_A \times \Sigma_B \times \Sigma_1 \times \ldots \times \Sigma_k \text{ and}$$
$$\Gamma = \Gamma_A \times \Gamma_B \times \Gamma_1 \times \ldots \times \Gamma_k$$

for some k, where the Σ_i and Γ_i represent the inputs and outputs respectively to the ith 'innocent' agent. We will call such a system a k-SDSFT.

We will now require that the k innocent agents choose distributions over strategies. An argument similar to Proposition 2 shows that we may assume that the innocent agents select deterministic strategies, and so we adopt a definition analagous to Definition 16.

Definition 32. *Let $\mathcal{T}$ be a k-SDFST. We define*

$$\mathcal{L}_n(\mathcal{T}) = \max_{\substack{x_B \in \mathcal{X}_B, \\ x_1 \in \mathcal{X}_1, \ldots, x_k \in \mathcal{X}_k}} \log \left| \left\{ y \in (\Sigma_B \times \Gamma_B)^k \middle| \exists x_A \in \mathcal{X}_A : \right.\right.$$
$$\left.\left. y \textit{ is consistent with } \mathcal{T}, x_A, x_B, x_1, \ldots, x_k \right\} \right|,$$

where $\mathcal{X}_i$ is the set of functions $(\Sigma_i \times \Gamma_i) \to \Sigma_i$, and consistency is defined similarly to Definition 12.

Our first open problem is to compute the min-entropy capacity of a k-SDFST. We conjecture that there should still be a dichotomy between polynomial and exponential growth.

C. Probabilistic systems

We may also wish to handle systems whose behaviour is probabilistic. We model such systems as probabilistic finite-state transducers.

Definition 33. *A* probabilistic finite-state transducer *is a tuple $\mathcal{T} = (Q, q_0, \Sigma, \Gamma, \Delta)$, where Q is a finite set of* states*, $q_0 \in Q$ is the* initial state *and $\Delta : Q \times \Sigma \times Q \times \Gamma \to \mathbb{R}^{\geq 0}$ is the* transition function*, such that for all $q \in Q$ and all $a \in \Sigma$ we have*

$$\sum_{b \in \Gamma, q' \in Q} \Delta(q, a, q', b) = 1.$$

We interpret $\Delta(q, a, q', b)$ as the probability that on receiving the input a in state q, the system outputs b and moves to state q'.

As before we require that Σ and Γ are of the form $\Sigma_A \times \Sigma_B$ and $\Gamma_A \times \Gamma_B$ respectively (although of course it would also be possible to consider multi-agent probabilistic systems), and the sets $\mathcal{X}_A$ and $\mathcal{X}_B$ are as before. For fixed x_A and x_B, the output Y produced to Bob after n steps is a sequence $y \in (\Sigma_B \times \Gamma_B)^n$, where $y = ((a'_1, b'_1), \ldots, (a'_n, b'_n))$ occurs with probability

$$\sum_{((a_1,b_1),\ldots,(a_n,b_n)) \in Z} \sum_{q_1,\ldots,q_n \in Q} \prod_{i=1}^{n} \Delta(q_{i-1}, (a_i, a'_i), q_i, (b_i, b'_i)),$$

where Z is the set of $((a_1, b_1), \ldots, (a_n, b_n)) \in (\Sigma_A \times \Gamma_A)^n$ such that $((a_1, a'_1), (b_1, b'_1)) \ldots ((a_n, a'_n), (b_n, b'_n))$ is consistent with x_A and x_B.

This defines an interactive channel $\mathcal{C}_n$, and so our second open problem is to compute the growth of $\mathcal{L}_\infty(\mathcal{C}_n)$. This seems to be a rather formidable challenge since we lack a way to reduce to a possibilistic view of Alice's actions, and so we may genuinely have to quantify over probability distributions for X_A.

V. Related work

So far as we are aware this is the first quantitative study which is able to analyse interactive systems in full generality, that is to say where inputs may be provided by both parties, according to distributions chosen adversarially so as to maximise information flow, rather than being specified as part of the system.

Mardziel, Alvim, Hicks and Clarkson in [16] consider interactive systems in essentially the same model as we use in Section III of this paper: they represent the system by a probabilistic finite automaton, which is executed in a 'context' consisting of the strategy functions for the high and low users. They then employ probabilistic programming to analyse particular systems with respect to particular contexts, demonstrating for instance that allowing an adaptive adversary can greatly increase information flow. However, they acknowledge that they are not able to analyse the maximum leakage over all possible contexts, instead observing that 'We consider such worst-case reasoning challenging future work'. The present work addresses this question for the case where the system is deterministic.

Köpf and Basin in [17] show how to calculate information leakage for a particular model relating to side-channel attacks in which the attacker is repeatedly permitted to make queries drawn from some fixed set. They give an exhaustive algorithm to compute the maximum amount of information leakege after n queries.

Boreale and Pampaloni in [18] consider the case of repeated queries issued by the attacker (possibly adaptively) to a stateless system and show that under certain reasonable assumptions the problem of computing the maximum leakage after n queries is NP-hard. In [19], the same authors together with Paolini study the asymptotics of the leakage resulting from n independent uses of a single channel for large n. This is in some sense dual to the situation we have considered, of the asymptotics of a single, long execution of a stateful system.

In [20], Andrés, Palamidessi, van Rossum and Smith compute the leakage of what they term 'interactive information-hiding systems' (IIHS), which are essentially automata over (secret) inputs and (observed) outputs. However, they assume an essentially passive attacker: apart from the values of the secret (whose distribution they sometimes allow to be chosen so as to maximise information flow), the system is assumed to follow known probabilistic behaviour. In follow-up work [21], Alvim, Andrés and Palamidessi demonstrate interesting connections between the mutual information capacity of such systems and the directed information capacity of channels with feedback, although this is of limited practical significance since it is now recognised that mutual information is not generally an appropriate measure of information flow.

An interesting alternative algorithmic approach is taken by Kawamoto and Given-Wilson in [22], although for a completely different problem from that addressed in this work. In [22], the authors consider a purely passive observer who is shown the outputs of two channels, interleaved according to some scheduler; the goal is to find a scheduler which minimises the information leakage. They show that this can be expressed as a linear programming problem, and therefore solved in time polynomial in the number of possible interleavings, which unfortunately is exponential in the number of possible traces.

VI. Conclusions

In [3], Ryan, McLean, Millen and Gligor write the following:

> Even at a theoretical level where timings are not available, and a bit per millisecond is not distinguishable from a bit per fortnight or a bit per century, a channel that compromises an unbounded amount of information is substantially different from one that cannot. Characterization of unbounded channels is suggested as the kind of goal that would advance the study of this subject

In Theorem 30 we have achieved this goal for deterministic systems, and in fact slightly more: we have shown that even among unbounded channels there is a dichotomy between 'safe' and 'dangerous' information flow, and this can be determined for a given system in polynomial time.

Having characterised the notion of safe versus dangerous information flow, one may ask about the question of enforcement of the safety criterion. In one sense this question is already answered by Theorem 30, since it includes a polynomial-time algorithm to determine whether the condition is satisfied for a given system. However, the development of automated tools implementing this algorithm, which preferably would allow realistic systems to be specified using more convenient notation than the rather abstract mathematical formalism of finite-state transducers, is certainly an important area for future work.

Acknowledgements

The author is grateful to Catuscia Palamidessi for helpful comments on an earlier version of this work, and to Dimiter Ostrev for comments on the final version.

References

[1] J. A. Goguen and J. Meseguer, "Security policies and security models," in *1982 IEEE Symposium on Security and Privacy*. IEEE, 1982, pp. 11–20.

[2] D. Mestel, "Widths of regular and context-free languages," *CoRR*, vol. abs/1709.08696, 2018. [Online]. Available: http://arxiv.org/abs/1709.08696

[3] P. Y. A. Ryan, J. McLean, J. Millen, and V. Gligor, "Non-interference, who needs it?" in *Proc. 14th IEEE Computer Security Foundations Workshop (CSFW '01)*, 2001, pp. 237–238.

[4] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, July 1948.

[5] G. Smith, "On the foundations of quantitative information flow," in *Proc. 12th Int. Conf. on Foundations of Software Science and Computational Structures (FOSSACS '09)*, 2009, pp. 288–302.

[6] M. S. Alvim, K. Chatzikokolakis, C. Palamidessi, and G. Smith, "Measuring information leakage using generalized gain functions," in *Proc. 25th IEEE Computer Security Foundations Symposium (CSF '12)*, June 2012, pp. 265–279.

[7] C. Dwork, "Differential privacy," in *Proc. 33rd International Conference on Automata, Languages and Programming (ICALP '06)*, 2006, pp. 1–12.

[8] M. S. Alvim, K. Chatzikokolakis, A. McIver, C. Morgan, C. Palamidessi, and G. Smith, "Additive and multiplicative notions of leakage, and their capacities," in *Proc. 27th IEEE Computer Security Foundations Symposium (CSF '14)*, July 2014, pp. 308–322.

[9] C. Braun, K. Chatzikokolakis, and C. Palamidessi, "Quantitative notions of leakage for one-try attacks," in *Proc. 25th Int. Conf. on Mathematical Foundations of Programming Semantics (MFPS '09)*, 2009, pp. 75 – 91.

[10] R. van der Meyden and C. Zhang, "A comparison of semantic models for noninterference," *Theoretical Computer Science*, vol. 411, no. 47, pp. 4123 – 4147, 2010.

[11] D. Clark and S. Hunt, "Non-interference for deterministic interactive programs," in *Formal Aspects of Security and Trust (FAST 2008)*, April 2009, pp. 50–66.

[12] G. H. Mealy, "A method for synthesizing sequential circuits," *Bell Labs Technical Journal*, vol. 34, no. 5, pp. 1045–1079, 1955.

[13] A. Roscoe, J. Woodcock, and L. Wulf, "Non-interference through determinism," in *Proc. 3rd European Symposium on Research in Computer Security (ESORICS '94)*, 1994, pp. 31–53.

[14] R. P. Dilworth, "A decomposition theorem for partially ordered sets," *Annals of Mathematics*, pp. 161–166, 1950.

[15] D. Mestel, "Quantifying information flow," Ph.D. dissertation, University of Oxford, 2018.

[16] P. Mardziel, M. S. Alvim, M. Hicks, and M. R. Clarkson, "Quantifying information flow for dynamic secrets," in *2014 IEEE Symposium on Security and Privacy*, May 2014, pp. 540–555.

[17] B. Köpf and D. Basin, "An information-theoretic model for adaptive side-channel attacks," in *Proc. 14th ACM Conference on Computer and Communications Security (CCS '07)*, 2007, pp. 286–296.

[18] M. Boreale and F. Pampaloni, "Quantitative information flow under generic leakage functions and adaptive adversaries," *Logical Methods in Computer Science*, vol. 11, no. 4, 2015.

[19] M. Boreale, F. Pampaloni, and M. Paolini, "Asymptotic information leakage under one-try attacks," in *Proc. 14th Int. Conf. on Foundations of Software Science and Computational Structures (FOSSACS '11)*, 2011, pp. 396–410.

[20] M. E. Andrés, C. Palamidessi, P. van Rossum, and G. Smith, "Computing the leakage of information-hiding systems," in *Proc. 16th Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems (TACAS '10)*, 2010, pp. 373–389.

[21] M. S. Alvim, M. E. Andrés, and C. Palamidessi, "Quantitative information flow in interactive systems," *Journal of Computer Security*, vol. 20, no. 1, pp. 3–50, Jan. 2012.

[22] Y. Kawamoto and T. Given-Wilson, “Quantitative information flow for scheduler-dependent systems,” in *Proc. 13th Workshop on Quantitative Aspects of Programming Languages and Systems (QAPL ’15)*, 2015, pp. 48–62.