Formal Verification of Differential Privacy in Concurrent Systems

Lili Xu

École Polytechnique

Thèse de Doctorat
Spécialité Informatique

Formal Verification of Differential Privacy in Concurrent Systems

Lili Xu

Supervisors
Konstantinos Chatzikokolakis
Huimin Lin
Catuscia Palamidessi

Defended on May 4th, 2015

École Polytechnique

Laboratoire d'Informatique

Équipe Comète

Abstract

The verification of systems for protecting sensitive and confidential information is becoming an increasingly important issue in the modern world. Many protocols for protecting confidential information have used randomized mechanisms, to obfuscate the link between the secret and the public information. Typical examples are DCNets, Crowds, Onion Routing, Freenet and Tor. Another common denominator of them is that various entities involved in the system to verify occur as concurrent processes, and present typically nondeterministic behavior.

This dissertation is devoted to the development of novel reasoning techniques for verifying differential privacy in concurrent systems. Differential privacy is a promising notion of privacy originated from the community of statistical databases, and now widely adopted in various models of computation. We use the principle of differential privacy as a criterion to measure the level of privacy that a concurrent system satisfies.

The first part of the present thesis is focused on modular reasoning about differential privacy in a probabilistic variant of Robin Milner's Calculus of Communicating Systems (CCS). We show that the calculus operators such as *non-deterministic choice*, *probabilistic choice*, *restriction* and a restricted form of *parallel composition* are safe, in the sense that combining components with these operators does not compromise the privacy of the entire system.

The second part focuses on the applicability of bisimulation - a fundamental technique in Concurrency Theory - for characterizing differentially private behavior. We borrow the idea of amortisation, which was initially applied on some bisimulations with cost-based actions, and coin an amortised probabilistic bisimulation. We show that it allows us to verify differential privacy and it is a more liberal notion than the work of Tschantz *et al.*

In the third part the focus is shifted to the development of proof systems - an axiomatic way for proving properties of concurrent systems. We provide sound and complete proof systems for our amor-

tised bisimulation and its weak counterpart. The proof systems make it possible to reason about long-term (observable) differentially private behavior by syntactic manipulation.

The last part presents an extension of the bisimulation metric based on the Kantorovich distance. This is a metric that has become very popular in Concurrency Theory, thanks to its principled and solid mathematical foundations. While the standard notion is additive in nature and therefore not suitable to prove the property of differential privacy (which is multiplicative), the extension developed in the thesis is parametric with respect to the underlying distance, and therefore suitable to capture a vast range of properties, including differential privacy.

Résumé

La vérification des systèmes de protection des données sensibles et confidentielles est un défi important que le monde moderne doive relever à l'ère du tout numérique. De nombreux protocoles pour la protection de ces données sensibles utilisent des mécanismes aléatoires pour masquer le lien entre une donnée secrète ou sensible et l'information publique qui lui est associée. Les systèmes de protection tels que DCNets, Crowds protocol, Onion Routing, Freenet et Tor sont des exemples typiques. L'autre dénominateur commun de ces systèmes est le fait que différentes entités impliquées dans le système se produisent simultanément, et présentent généralement un comportement non-déterministe.

Cette thèse est consacrée au développement des nouvelles techniques de raisonnement pour vérifier le concept de la "differential privacy" dans les systèmes concurrents. Differential privacy est une notion prometteuse provenant de la communauté des bases statistiques. Elle est maintenant largement adoptée dans différents modèles de calcul. Nous l'utilisons dans cette thèse comme un critère pour mesurer le niveau de confidentialité qu'un système probabiliste et non-déterministe satisfait.

Dans la première partie de cette thèse, nous considérons le raisonnement modulaire sur la differential privacy dans une variante probabiliste de "Calcul des Systèmes Communicants" (CCS) de Robin Milner. Nous montrons que les opérateurs du modèle de calcul tels que le *choix non-déterministe*, le *choix probabiliste*, la *restriction* ainsi qu'une forme restreinte de la *composition parallèle* préservent tous la differential privacy, dans le sens que combiner des composantes (ayant un certain niveau de protection au sens de la differential privacy) avec ces opérateurs ne compromet pas la differential privacy du système entier ainsi obtenu.

La deuxième partie porte sur l'applicabilité de bisimulation - une technique fondamentale dans le domaine de la concurrence - pour caractériser le comportement du système au sens de la differential privacy. Nous adoptons l'idée de l'amortissement, qui a d'abord été

appliquée sur certaines bisimulations avec des actions en fonction des coûts, et construisons une bisimulation probabiliste amortie. Cette bisimulation nous a permis de vérifier la differential privacy. Nous avons également montré que notre notion est plus flexible que celle dans le travail de Tschantz *et al.*

Dans la troisième partie l'attention porte sur le développement des systèmes de preuve : une manière axiomatique pour prouver les propriétés de systèmes concurrents. Nous fournissons deux systèmes de preuve cohérents et complets pour notre bisimulation amortie et son homologue (c'est-à-dire la bisimulation) faible. Les systèmes de preuve permettent de raisonner sur le comportement (observable) sur le long terme d'un système grâce à la manipulation syntaxique de son modèle.

La dernière partie présente une extension de la métrique de bisimulation basée sur la distance de Kantorovich. C'est une mesure qui est devenue très populaire dans le domaine de la concurrence, grâce à ses fondements mathématiques solides. Mais la notion standard est de nature additive et donc pas appropriée à prouver la propriété de la differential privacy (qui est de nature multiplicative), l'extension développée dans la thèse est paramétrique par rapport à la distance sous-jacente, et donc appropriée pour capturer une vaste gamme de propriétés, y compris la differential privacy.

Acknowledgements

I am deeply grateful to my supervisors, Huimin Lin and Catuscia Palamidessi. It is the greatest honor in my life to be supervised by two such remarkable computer scientists. Huimin is a genuine researcher. His faith and persistence in the pursuit of truth lets me know a basic principle a researcher should have. Thanks to him for always listening to my ideas and guiding me with his broad view in concurrency theory. Catusica is a widely recognized researcher, a patient supervisor and also a wonderful woman. She understands how important it is to first build confidence in research, encouraging me to address simple issues at the beginning, to give presentations and attend summer schools for scientific publicity. She taught me how to write papers, read my drafts and gave a lot suggestions for improvement. Every discussion that we had and the nights we worked together are meaningful moments for me. Also many thanks to the fantastic social events held by her that made my spiritual growth in understanding various cultures. Her brilliance, generosity and kindness have set an example that I will always keep with me in academia and my life.

I would like to express my gratitude to my co-authors Kostas Chatzikokolakis, Sardaouna Hamadou and Daniel Gebler. I was deeply impressed and influenced by their great ability to find problems and to suggest elegant solutions. I would also like to thank Frank Valencia who shared with me many useful research written and presentation skills.

I would like to seize this opportunity to thank the people that I met in the wonderful team Comète. They are Mario Alvim, Miguel E. Andrés, Ryuta Arisaka, Andrés Aristizabal, Nico Bordenabe, Matteo Cimini, Ehab ElSalamouny, Thomas Given-Wilson, Tobias Heindel, Yusuke Kawamoto,

Sophia Knight, Luis Pino and Marco Stronati. Thank you for helping me to settle down in Paris, and moreover, for sprinkling my life with interesting and meaningful discussions on cultural stories, social and political events. Thanks to my friends Songzhe Han, Kailiang Ji, Lin Qin, Xiao Wang, Qian Wang and Jie Yang who reduced my homesickness by offering me delicious Chinese food and joyful events. And many thanks to Valérie Berthou, Marie-Jeanne Gaffard and Christelle Lievin for providing important administrative support for my stay in France.

Finally, I would like to thank my family, and most of all my husband You Tang. I couldn't have done this without your endless support.

Contents

List of Figures

One

Introduction

The most recent developments and usages of information technologies such as data profiling in databases, or user tracking in pervasive computing, pose serious threats to the confidential information of the users. For instance, the social networks Twitter and Flickr carefully protect their users' data by anonymization, and yet Narayanan and Smatikov [NS09] were able to conceive a de-anonymization algorithm which could re-identify 30% of the people who have accounts in both of them, with only a 12% error rate. The verification of systems for protecting sensitive and confidential information is becoming an increasingly important issue in the modern world.

This thesis is devoted to the formal verification of differential privacy in probabilistic concurrent systems. Differential privacy is a promising notion of privacy originated from the community of statistical databases, and now widely adopted in various models of computation. We shall use the principle of differential privacy as a criterion to measure the level of privacy that a concurrent system satisfies.

1.1 Concurrent and Probabilistic Processes

Many protocols for protecting confidential information have been proposed in the literature. In order to obfuscate the link between the secret and the public information, several of them use randomized mechanisms. Typical

examples are DCNets [Cha88], Crowds [RR98], Onion Routing [SGR97], Freenet [CSWH00] and Tor [DMS04]. Another common denominator is that various entities involved in the system to verify occur as concurrent processes, and present typically nondeterministic behavior. A formal apparatus for reasoning about these systems should allow to characterize both non-deterministic and probabilistic behaviors.

Process algebras, also known as process calculi, are a powerful mathematical model for the specification and verification of concurrent systems. They provide a formalism for representing and reasoning about the behaviors of distributed systems, algorithms and protocols (in a compositional way). Some of the most prominent representants of these formalisms are CCS [Mil89], ACP [BK84, BW90] and CSP [Hoa85].

In a process algebra, typically there are only a few operators, such as action prefix, summation (nondeterministic choice), recursion and parallel composition. The latter is particularly important for concurrency, since it allows to specify the structure of systems composed of several interacting agents. An internal action, τ, which is one of the most important features in the design of process algebra, is used to represent synchronization between agents. In this thesis we focus on CCS, because it allows all of these kinds of behaviors but still is simple and general.

Bisimulation is a central notion at the heart of the theory of process calculi [Mil89]. The idea is to match any transition in one process with a transition labelled by the same action in the other process, and their residuals can continue to mimic each other. Weak bisimulation relaxes bisimulation by regarding two systems as equivalent if they exhibit the same pattern of *external* actions. Weak bisimulation is proved fundamental for the verification of systems where abstraction from internal actions is essential. The correctness of a system can then be verified by proving that it is equivalent to its desired external behavior, which can be expressed as process terms. (Weak) Bisimulation equivalence is a mathematically elegant, tractable concept. Many tools, either automated or interactive, have been developed for proving bisimulation between processes (e.g. [CAD, Lin95]). The principle of bisimulation will also play an important role in

this thesis, being used to reason about behaviors of systems.

The theory of process algebra has been applied to systems equipped with quantitative features, such as cost [KAK05], time [LY00] and probabilities [BS01]. For specifying probabilistic behaviors in security protocols and systems, in this thesis we focus on probabilistic systems. In [vGSS95], van Glabbeek *et al.* classified probabilistic models into *reactive, generative* and *stratified*. In [Seg95] Segala pointed out that neither reactive nor generative nor stratified models capture real nondeterminism. He then introduced a model, the *probabilistic automata* (PA), where both probability and nondeterminism are taken into account. Segala further proposed a simplified version of PA called *simple probabilistic automata* (SPA), which are similar to ordinary automata except that a labelled transition leads to a probabilistic distribution over a set of states instead of a single state. In this thesis, we shall use SPA as the operational semantics of our process algebra.

For probabilistic systems, accordingly, a notion of *probabilistic bisimulation* was first defined in [LS91]. It has been admitted to be not *robust*, namely small changes to any of those probabilities may cause equivalent states to become inequivalent. This is particularly relevant for security systems where requiring all agents to behave identically is impractical. It is therefore desirable to know the extent they differ from each other, thus seeing how difficult it is for the attackers to differentiate them. This started the quest for approximate notions of behavioral equivalence for probabilistic systems.

Originally proposed in the seminal works of van Breugel and Worrel [vBW01b, vBW01a] and of Desharnais *et al.* [DGJP99, DJGP02, DJGP04], the pseudometric based on the Kantorovich lifting has become very popular in the process algebra community. It is particularly appealing because it extends weak bisimilarity (captured by the property of having distance 0) and it is based on a natural way of relating probability masses distributed on a metric space. More recently, the Kantorovich bisimilarity metric has been shown to provide a bound on the statistical distance on probabilistic traces [CvBW12]. This means that it can be used to verify certain proba-

bilistic properties on traces. More specifically, these properties are linear, in the sense that the difference increases linearly wrt variations on the distributions. Whether and how the solid theoretical background built for the pseudometric based on the Kantorovich lifting can be explored to characterize properties other than linear ones motivates part of the work of the thesis.

1.2 Differential Privacy

Several formalization of the notion of protection have been proposed in the literature. Among those based on probability theory, we mention the true-or-false properties in the *Anonymity* hierarchy: like *strong anonymity* which describes an ideal situation where a protocol does not leak any information about the identity of the user, and some weaker notions like *conditional anonymity* [Cha88, HO05, BP05] and *probable innocence* [RR98]. More refined approaches, based on information theory, aim at measuring also the degree of protection provided by a system. The idea is to express the leakage of information in terms of the notion of mutual information. A nice feature of this approach is that we can consider different notions of entropy depending on the kind of adversary we want to model. The most used are the Shannon entropy, see for example [CHM05, Mal07], and the Rényi min-entropy [Smi09].

Differential privacy [Dwo06, DL09, Dwo11] is a promising definition of confidentiality that has emerged recently from the field of statistical databases. It provides strong privacy guarantees, and requires fewer assumptions than the information-theoretical approach. We say that a system is ϵ-differentially private if for every pair of *adjacent* datasets (i.e. datasets which differ in the data of an individual only), the probabilities of obtaining a certain answer differ at most by a factor e^{ϵ}. Differential privacy captures the intuitive requirement that the (public) answer to a query should not be affected too much by the (private) data of each singular individual.

Although differential privacy originates from the field of statistical databases, it is becoming increasingly popular in many other fields, ranging from

programming languages [RP10] to social networks [NS09] and geolocation [MKA+08]. One of the reasons of its success is its independence from side knowledge, which makes it robust to attacks based on combining various sources of information.

The extension of the principle of differential privacy to secrets with a generic adjacency relation has been universally adopted in the literature, see for instance [BKOB12, GHH+13, CABP13]. Therein, the sensitive information to be protected was other than the value of a single individual in databases. A general notion of adjacency was considered, formalized by a general metric that measures the distance between values of secrets. They showed that within this setting, it is still reasonable to employ the same principle of differential privacy and to obtain a meaningful notion of privacy. In this thesis we also consider the principle of differential privacy under a general notion of adjacency, which provides the basis for formalizing also other security concepts, like anonymity.

1.3 This thesis: Differential Privacy in Concurrent Systems

The goal of this thesis is to develop formalisms for probabilistic concurrent systems that can reason about differentially private behaviors. We address this issue mainly from three directions: modular reasoning provided by process combinators, distance measuring based on approximate bisimulations and axiomatic theories.

1.3.1 Modular reasoning

In Chapter 3 we consider a probabilistic process calculus equipped with secret labels as a specification formalism for concurrent systems, and we propose a framework for reasoning about the degree of differential privacy provided by such systems. In particular, we investigate the preservation of the degree of privacy under composition via the various operators. We illustrate our idea using a variant of the anonymity protocol: Crowds. Our

extension allows anonymous users to send messages probabilistically over the users they think trustable, rather than over all users as strictly required in standard Crowds. Furthermore, we show that this trust information may compromise privacy, and we introduce a notion of adjacency relation to eliminate this factor, thus retrieving the real level of differential privacy in this case. Then, we investigate how the users' preference levels affect the degree of privacy.

1.3.2 Bisimulations for differential privacy

In Chapter 4 we investigate techniques for proving differential privacy based on approximate bisimulations. Our motivation stems from the work of Tschantz *et al.* [TKD11], who proposed a verification method based on proving the existence of a stratified family between states, that can track the privacy leakage, ensuring that it does not exceed a given leakage budget. We improve this technique by investigating a state property which is more permissive and still implies differential privacy. We propose a new probabilistic bisimulation by integrating the notion of *amortisation*, which results into a more parsimonious use of the privacy budget. We show that the closeness of automata in our amortised bisimulation still guarantees the preservation of differential privacy, which makes it suitable for verification. Moreover we show that our amortised bisimulation is substitutive under typical process combinators. We apply the bisimulation verification framework to reason about the degree of differential privacy of protocols by the example of the Dining Cryptographers Protocol with biased coins.

1.3.3 Complete proof systems

The concept of amortisation was initially introduced for cost-based bisimulations [KAK05, dFERVGR07] to make long-term behavioral comparisons between nondeterministic systems. The idea of amortisation has been borrowed in the previous chapter to formulate an approximate notion for probabilistically behavioral equivalence: *amortised probabilistic bisimulation*. In Chapter 5 we present sound and complete proof systems for amortised

strong probabilistic bisimulation and its weak counterpart, and prove their soundness and completeness. Our results make it possible to reason about long-term (observable) probabilistic behaviors by syntactic manipulation.

1.3.4 Generalized bisimulation metrics

The pseudometric based on the Kantorovich lifting is one of the most popular notions of distance between probabilistic processes proposed in the literature. However, its application in verification is limited to linear properties. In Chapter 6 we propose a generalization which allows to deal with a wide class of properties, such as those used in security and privacy. More precisely, we propose a family of pseudometrics, parameterized on a notion of distance which depends on the property we want to verify. Furthermore, we show that the members of this family still characterize bisimilarity in terms of their kernel, and provide a bound on the corresponding distance between trace distributions. Then we study the instance corresponding to differential privacy, and we show that it has a dual form, easier to compute. We also prove that the typical process-algebra constructs are non-expansive, thus paving the way to a modular approach to verification.

Related works

We present some related work concerning verification of differential privacy in various contexts, and formalisms of other notions of information protection based on process calculi. Detailed comparison between our work of each part and works in the literature can be found sprinkled in each chapter. To the best of our knowledge, the line of work in this thesis is the first to investigate differential privacy for concurrent systems within the setting of process calculi.

Verification of differential privacy has been itself an active area of research. It has been investigated in a SQL-like language [McS09] for statistical databases and a MapReduce-based system for cloud computing [RSK+10]. Prominent approaches based on formal methods are those based on type systems [RP10, GHH+13] and logical formulations [BKOB12, BDG+13].

An earlier paper [TKD11] defined stratified bisimulation relations suitable for proving differential privacy, however it suffered from the fact that the respective kernel relations do not fully characterize probabilistic bisimilarity.

Among several formalizations of the notion of information protection based on probability theory, we mention some rather popular approaches, mainly based on information theory, in particular, to consider different notions of entropy depending on the kind of adversary, and to express the leakage of information in terms of the notion of mutual information. We name a few works also discussed in the models of probabilistic automata and process algebra: Boreale [Bor06] establishs a framework for quantifying information leakage using absolute leakage, and introduces a notion of rate of leakage. Deng *et al.* [DPW06] use the notion of relative entropy to measure the degree of anonymity. Compositional methods based on Bayes risk method are discussed by Braun *et al.* [BCP08, CPB]. A metric for probabilistic processes based on the Jensen-Shannon divergence is proposed in [Mu09] for measuring information flow in reactive processes. Unlike the information-theoretical approach, differential privacy provides strong privacy guarantees independently from side knowledge. However, progress for studying differential privacy in the models of probabilistic automata and process algebra has been relatively new. It would be interesting to see how the issues stressed and the reasoning techniques developed there can be adapted for differential privacy.

1.4 Plan of the Thesis and Contributions

This thesis provides the theoretical basis for developing algorithms and tools of verifying or computing differential privacy in probabilistic processes. The most significant contributions of this thesis can be summarized as follows:

1. We prove that non-deterministic choice, probabilistic choice, the restriction operator and a restricted form of parallel composition preserves differential privacy, in the sense that combining components

under these operators does not compromise the privacy of the system (in Chapter 3).

2. We propose an approximate notion for probabilistically behavioral equivalence: amortised probabilistic bisimulation. We show that this bisimulation is suitable for verifying differential privacy and it improves the method based on a stratified family between states in the work of Tschantz *et al.*(in Chapter 4).

   Furthermore, we build sound and complete proof systems for amortised probabilistic bisimulation and its weak counterpart, and prove their soundness and completeness, which makes it possible to reason about differentially private behaviors by syntactic manipulation. (in Chapter 5).

3. We propose a family of generalized bisimulation metrics based on the Kantorovich lifting. The kernel of this family fully characterized bisimilarity. The metrical closeness between processes indicates a bound on the distance between the corresponding trace distributions. This allows to deal with a wide class of behavioral properties; importantly, one of its instances corresponds to differential privacy (in Chapter 6).

Besides these four chapters, there are two introductory chapters, the first being the present introduction. Chapter 2 introduces some preliminary information about probability spaces, probabilistic automata, a probabilistic version of CCS and differential privacy. Finally, in Chapter 7 we make our final conclusions.

1.5 Publications

Most of the results in this thesis have already appeared in scientific publications. More specifically:

- Chapter 3 is based on the paper **Modular Reasoning about Differential Privacy in a Probabilistic Process Calculus** [Xu12]

that appeared in the proceedings of the 7^{th} *International Symposium on Trustworthy Global Computing* (TGC 2012), and the complementary technical report **Privacy-Preserving Process Constructors** [XHP14].

- Chapter 4 is based on the paper **Metrics for Differential Privacy in Concurrent Systems** [XCL14], which was published in the 34^{th} *IFIP WG 6.1 International Conference on Formal Techniques for Distributed Objects, Components and Systems* (FORTE 2014).

- Chapter 5 is based on the paper **Complete Proof Systems for Amortised Probabilistic Bisimulations** [XL14], which is about to appear on *Journal of Computer Science and Technology*.

- Chapter 6 is based on the paper **Generalized Bisimulation Metrics** [CGPX14] that appeared in the proceedings of the 25^{th} *International Conference on Concurrency Theory* (CONCUR 2014).

Two

Preliminaries

In this chapter we give a brief overview of the technical concepts that will be used throughout this thesis.

2.1 Probability Spaces

In this section we recall some basic notions in Probability Theory.

Let X be a set. A *σ-field* (or σ-algebra) over X is a collection $\mathcal{F}$ of subsets of X closed under complement and countable union and such that $X \in \mathcal{F}$.

Definition 2.1.1 (Probability measure)**.** *A* probability measure *on $\mathcal{F}$ is a function $\mu : \mathcal{F} \to [0, \infty)$ such that*

1. $\mu(\emptyset) = 0$*;*
2. $\mu(\bigcup_i C_i) = \sum_i \mu(C_i)$*, where $\{C_i\}_i$ is a countable collection of pairwise disjoint elements of $\mathcal{F}$;*
3. $\mu(X) = 1$.

Definition 2.1.2 (Probability space)**.** *A* probability space *is a tuple $(X, \mathcal{F}, \mu)$ where*

1. *X is a set, called the* sample space*;*

2. *$\mathcal{F}$ is a σ-field on X called the* event space*, of which elements are called* events*;*

3. *μ is a probability measure on $\mathcal{F}$.*

A probability space and the corresponding probability measure are called *discrete*, if $\mathcal{F} = 2^X$ and

$$\forall C \in \mathcal{F}.\ \ \mu(C) = \sum_{x \in C} \mu(\{x\})$$

In this case, we can construct μ from a function $p : X \to [0,1]$ satisfying $\sum_{x \in X} p(x) = 1$ by assigning $\mu(\{x\}) = p(x)$. The function p is called a *probability distribution* over X. We denote by $Prob(X), Disc(X)$ the set of all and discrete probability measures (or distributions) over X respectively. Given $x \in X$, we use $\delta(x)$, (called the *Dirac measure* on x), to denote the probability distribution that assigns 1 to the event $\{x\}$, namely, if $y = x$ then $\delta(x)(\{y\}) = 1$, otherwise $\delta(x)(\{y\}) = 0$. The set $\{x \in X \mid \mu(x) > 0\}$ is called the *support set* of μ, represented by $supp(\mu)$.

Definition 2.1.3 (Conditional probability)**.** *If A and B are two events, then their joint $A \cap B$ is also an event. If $\mu(A) > 0$, then we define the* conditional probability $p(B \mid A)$*, representing the probability of B given that A holds, as*

$$p(B \mid A) \stackrel{def}{=} \frac{\mu(A \cap B)}{\mu(A)}$$

If $p(B \mid A) = \mu(B)$, namely $\mu(A \cap B) = \mu(A) \cdot \mu(B)$, then A and B are *independent*. If $p(A \cap B \mid C) = p(A \mid C) \cdot p(B \mid C)$, then A and B are *conditionally independent* given that C holds.

2.2 Probabilistic Automata

We recall here some basic concepts about probabilistic automata, following the notions of *simple* probabilistic automata in [Seg95]. It will be used as the operational semantics of probabilistic process algebra.

Definition 2.2.1 (Probabilistic automata)**.** *A* probabilistic automaton *(henceforth PA)*[1] $\mathcal{A}$ *is a tuple* $(S, \overline{s}, A, D)$ *where*

- S *is a countable set of states;*
- $\overline{s} \in S$ *is the start state;*
- A *is a finite set of labels;*
- $D \subseteq S \times A \times Disc(S)$ *is a* transition relation.

Informally, if $(s, a, \mu) \in D$ then there is a transition from the state s performing a label a and then leading to a distribution μ over a set of states instead of a single state. It is also occasionally written as $s \xrightarrow{a} \mu$. The transition in D is chosen nondeterministically, and the target state among the ones allowed by μ is chosen probabilistically.

A *fully probabilistic automaton* (henceforth FPA) is a probabilistic automaton without nondeterminism, namely at each state at most one transition can be chosen. We denote by $L(s)$ and $\pi(s)$ the label and distribution of the unique transition starting from s (if any).

We say that a PA $\mathcal{A}$ is *finitely branching* iff the nondeterministic choices at any state are finite, and $supp(\mu)$ is finite for all $s \xrightarrow{a} \mu$.

An *execution* α of a probabilistic automaton is a (possibly infinite) sequence of alternating states and labels $s_0 a_0 s_1 a_1 s_2 a_2 s_3 \cdots$, such that for each i there is a transition $(s_i, a_i, \mu_i) \in D$ with $\mu_i(s_{i+1}) > 0$. We will use $Exec^*(\mathcal{A})$ to represent the set of all the finite executions of $\mathcal{A}$, $Exec(\mathcal{A})$ to represent the set of all the executions of $\mathcal{A}$, and $lstate(\alpha)$ to denote the last state of a finite execution $\alpha \in Exec^*(\mathcal{A})$. $X \rightharpoonup Y$ represents the partial functions from X to Y.

Definition 2.2.2 (Scheduler)**.** *A* scheduler *of a probabilistic automation* $\mathcal{A} = (S, \overline{s}, A, D)$ *is a function*

$$\zeta : Exec^*(\mathcal{A}) \rightharpoonup D$$

such that $\zeta(\alpha) = (s, a, \mu) \in D$ *implies that* $s = lstate(\alpha)$.

[1] PA defined here is in fact the "simple probabilistic automata" defined in [Seg95, SL94]. For simplicity we just call them "probabilistic automata" in the rest of the thesis.

Intuitively, a scheduler resolves the nondeterminism by selecting a transition among the ones available in D, based on the history of the execution.

Definition 2.2.3 (Execution tree)**.** *The* execution tree *of* $\mathcal{A}$ with respect to a scheduler ζ*, denoted by* $etree(\mathcal{A}, \zeta)$*, is a fully probabilistic automaton* $\mathcal{A}' = (S', \overline{s}', A', D')$ *satisfying*

1. $S' = Exec(\mathcal{A})$;
2. $\overline{s}' = \overline{s}$;
3. $A' = A$;
4. $(\alpha, a, \mu') \in D'$ *if there exists a distribution* μ *such that* $\zeta(\alpha) = (lstate(\alpha), a, \mu)$ *and* $\mu'(\alpha a s) = \mu(s)$.

Intuitively, $etree(\mathcal{A}, \zeta)$ is produced by unfolding the executions of $\mathcal{A}$ and resolving all nondeterministic choices using ζ.

A *trace* is a sequence of labels in $A^* \cup A^\omega$ obtained from executions by removing the states. We use $[\,]$ to represent the empty trace, and $\frown$ to concatenate two traces. In a FPA $\mathcal{A}$, a state s of $\mathcal{A}$ induces a probability measure over traces as follows. The basic measurable events are the cones of finite traces, where the cone of a finite trace $\vec{t}$, denoted by $C_{\vec{t}}$, is the set $\{\vec{t'} \in A^* \cup A^\omega | \vec{t} \leq \vec{t'}\}$, where $\leq$ is the standard prefix preorder on sequences. The probability induced by s on a cone $C_{\vec{t}}$, denoted by $\Pr[s \rhd C_{\vec{t}}\,]$, is defined recursively as follows:

$$\Pr[s \rhd C_{\vec{t}}\,] = \begin{cases} 1 & \text{if } \vec{t} = [\,] \\ 0 & \text{if } \vec{t} = a \frown \vec{t'} \text{ and } a \neq L(s) \\ \sum_{s_i} \mu(s_i) \Pr[s_i \rhd C_{\vec{t'}}\,] & \text{if } \vec{t} = a \frown \vec{t'} \text{ and } s \xrightarrow{a} \mu \end{cases} \quad (2.1)$$

For simplicity reasons, we write $\sum_{s_i}$ instead of $\sum_{s_i \in S}$ whenever the summation is taken over the default range S. This probability measure is extended to arbitrary measurable sets in the σ-algebra of traces in the standard way. We write $\Pr[s \rhd \sigma]$ to represent the probability induced by s on the measurable set of traces σ.

There are other probabilistic models analogous to ours, for example, the reactive probabilistic automata used in [LS89, vGSS95] and the alternating model defined in [HJ90]. The former equips a transition with the information of both an action label and a probability, which would not make intuitive the distinction between non-deterministic and probabilistic behaviors. The latter classifies the set of states into two categories: one for non-deterministic states and the other for probabilistic states, where an execution of a system is generated by a strict alternation of the two kinds of states. From the point of view of expressiveness, these two kinds of models and our model are basically the same and can be converted to each other (see the survey [SV04], and the difference between the alternating and non-alternating models in axiomatizations in [BS01]).

2.3 Probabilistic Process Algebra

In this section we present a probabilistic process algebra CCS_p. It extends Milner's CCS [Mil89] with an addition of a probabilistic choice operator, allowing to describe both non-deterministic and probabilistic behaviors.

Let I be a finite set of indices, a be an element of a countable set of *channel names*. The syntax of CCS_p is:

$\alpha ::=$	$a \mid \overline{a} \mid \tau$		*prefixes*
$Q, R ::=$			*process term*
	$\mathbf{0}$		*void process*
$\mid$	$\alpha.Q$		*prefixes*
$\mid$	$\bigoplus_{i \in I} p_i Q_i$		*probabilistic choice*
$\mid$	$Q + R$		*non-deterministic choice*
$\mid$	$Q \mid R$		*parallel composition*
$\mid$	$(\nu a)Q$		*restriction*
$\mid$	$!Q$		*replication*

The term $\mathbf{0}$, representing the terminated process, is syntactic sugar for an empty non-deterministic choice. The $+$ operator is the classical nonde-

$$\text{ACT}\quad \frac{}{\alpha.Q \xrightarrow{\alpha} \delta(Q)} \qquad \text{SUM1}\quad \frac{Q \xrightarrow{\alpha} \mu}{Q + R \xrightarrow{\alpha} \mu}$$

$$\text{PROB}\quad \frac{}{\bigoplus_{i\in I} p_i\, Q_i \xrightarrow{\tau} \sum_i p_i\, Q_i} \qquad \text{PAR1}\quad \frac{Q \xrightarrow{\alpha} \mu}{Q \,|\, R \xrightarrow{\alpha} \mu \,|\, R}$$

$$\text{COM}\quad \frac{Q \xrightarrow{a} \delta(Q') \quad R \xrightarrow{\overline{a}} \delta(R')}{Q \,|\, R \xrightarrow{\tau} \delta(Q' \,|\, R')} \qquad \text{RES}\quad \frac{Q \xrightarrow{\alpha} \mu \qquad \alpha \neq a, \overline{a}}{(\nu a)Q \xrightarrow{\alpha} (\nu a)\mu}$$

$$\text{REP1}\quad \frac{Q \xrightarrow{a} \mu}{!\,Q \xrightarrow{a} \mu \,|\, !\,Q} \qquad \text{REP2}\quad \frac{Q \xrightarrow{a} \delta(Q_1) \quad Q \xrightarrow{\overline{a}} \delta(Q_2)}{!\,Q \xrightarrow{\tau} \delta(Q_1 \,|\, Q_2 \,|\, !\,Q)}$$

Figure 2.1: The semantics of CCS_p

terministic choice as defined in [Mil89]. It is folklore that $+$ is associative and commutative. We shall write $\sum_{i\in 1..n} Q_i$ for $Q_1 + Q_2 + \cdots + Q_n$.

The term $\bigoplus_{i\in I} p_i Q_i$ represents a *blind probabilistic choice*, where p_i's satisfy $p_i \in (0, 1]$ and $\sum_{i\in I} p_i = 1$. We use the notation $Q_1 \oplus_p Q_2$ to represent a binary sum with $p_1 = p$ and $p_2 = 1 - p$. When $I = \{1\}$ it will be abbreviate to $\Delta(Q_1)$.

We let μ, ν range over distributions over processes. We represent a distribution over processes by $\mu = \sum_{i\in I} p_i Q_i$ where $\mu(Q_i) = p_i$. When $n = 1$, it is degenerated to the Dirac measure on Q_1, namely, $\mu = \delta(Q_1)$.

The operational semantics of a CCS_p term Q is a probabilistic automaton whose states are the processes reachable from Q, and whose transition relation is defined according to the rules in Fig. 2.1. SUM1 and PAR1 have corresponding right rules SUM2 and PAR2, omitted for simplicity. We use $Q \xrightarrow{a} \mu$ to represent the transition (Q, a, μ). We denote by $\mu \,|\, R$ the measure μ' such that $\mu'(Q \,|\, R) = \mu(Q)$ for all processes Q, and $\mu'(Q') = 0$ if Q' is not of the form $(Q \,|\, R)$. Similarly $(\nu a)\mu = \mu'$ such that $\mu'((\nu a)Q) = \mu(Q)$.

A transition of the form $Q \xrightarrow{a} \delta(Q')$, having for target a Dirac measure, corresponds to a transition of a non-probabilistic automaton. All the rules in Fig. 2.1 follow the transition rules in standard CCS, except the rule PROB. The latter models the internal probabilistic choice: a silent

τ transition is available from the sum to a measure containing all of its operands, with the corresponding probabilities. Note that in the produced probabilistic automaton, all transitions to non-Dirac measures are silent.

2.4 Probabilistic bisimilarity

We recall the notion of probabilistic bisimilarity first defined in [LS91].

An equivalence relation over S can be lifted to a relation over distributions over S by stating that two distributions are equivalent if they assign the same probability to the same equivalence class. Formally, given an equivalence relation $\mathcal{R}$ on S, its lifting $\mathcal{L}(\mathcal{R})$ is an equivalence relation on $Disc(S)$, defined as

$$(\mu, \mu') \in \mathcal{L}(\mathcal{R}) \quad \text{iff} \quad \forall s \in S : \mu([s]_{\mathcal{R}}) = \mu'([s]_{\mathcal{R}})$$

where $[s]_{\mathcal{R}}$ denotes the equivalence class of s wrt $\mathcal{R}$.

We recall the notions of probabilistic bisimulation and bisimilarity, following the formulation in terms of post-fixpoints of a transformation on state relations:

Definition 2.4.1.

- *The transformation $B : S \times S \to S \times S$ is defined as: $(s, s') \in B(\mathcal{R})$ iff*
  - *if $s \xrightarrow{a} \mu$, then there exists μ' such that $t \xrightarrow{a} \mu'$ and $(\mu, \mu') \in \mathcal{L}(\mathcal{R})$,*
  - *if $s' \xrightarrow{a} \mu'$, then there exists μ such that $s \xrightarrow{a} \mu$ and $(\mu', \mu) \in \mathcal{L}(\mathcal{R})$.*
- *A relation $\mathcal{R} \subseteq S \times S$ is called a bisimulation if it is a post-fixpoint of $\mathcal{R}$, i.e. $\mathcal{R} \subseteq B(\mathcal{R})$.*

It is easy to see that B is monotonic on $(2^{S \times S}, \subseteq)$ and that the latter is a complete lattice, hence by Tarski's theorem there exists the greatest fixpoint of B, and it coincides with the greatest bisimulation:

Definition 2.4.2. *The bisimilarity relation $\sim \subseteq S \times S$ is defined as:*

$$\sim \;=\; \max\{\mathcal{R} \mid \mathcal{R} = B(\mathcal{R})\} \;=\; \max\{\mathcal{R} \mid \mathcal{R} \subseteq B(\mathcal{R})\} \;=\; \bigcup\{\mathcal{R} \mid \mathcal{R} \subseteq B(\mathcal{R})\}$$

2.5 Pseudometrics

A pseudometric is a relaxed notion of a normal metric in which distinct elements can have distance zero. We consider here a generalized notion where the distance can also be infinite, and we use $[0, +\infty)$ to denote the non-negative fragment of the real numbers $\mathbb{R}$ enriched with $+\infty$. Formally, an (extended) *pseudometric* on a set X is a function $m : X^2 \to [0, +\infty)$ with the following properties:

- (reflexivity) $m(x, x) = 0$,
- (symmetry) $m(x, y) = m(y, x)$,
- (triangle inequality) $m(x, y) \leq m(x, z) + m(z, y)$.

A *metric* has the extra condition that

$$m(x, y) = 0 \text{ implies } x = y$$

Let $\mathcal{M}_X$ denote the set of all pseudometrics on X with the ordering $m_1 \preceq m_2$ iff $\forall x, y. m_1(x, y) \leq m_2(x, y)$. It can be shown that $(\mathcal{M}_X, \preceq)$ is a complete lattice with bottom element $\bot$ such that $\forall x, y. \bot(x, y) = 0$ and top element $\top$ such that $\forall x, y. \top(x, y) = +\infty$ if $x \neq y$ and 0 otherwise.

2.6 Differential Privacy

Differential Privacy [Dwo06] captures the idea that a query on a dataset does not provide too much information about a particular individual, regardless of whether the individual's record is in the dataset or not. In order to achieve this goal, typically some probabilistic noise is added to the answer. The formal definition is the following (where $\mathcal{M}$ denotes the randomized answer, Pr the probability measure, and ϵ a finite non-negative number): Two datasets x_1 and x_2 are adjacent if their *Hamming distance* is not greater than 1, namely, they are differing in only one record, i.e. a single row.

Definition 2.6.1 (Differential Privacy [Dwo06])**.** *A mechanism $\mathcal{M}$ provides* ϵ-differential privacy *iff for all adjacent datasets x_1 and x_2, and for all $Z \subseteq Range(\mathcal{M})$,*

$$\Pr[\mathcal{M}(x_1) \in Z] \leq e^{\epsilon} \cdot \Pr[\mathcal{M}(x_2) \in Z]$$

It has been shown in [McS09] that, when the composition properties of differential privacy is studied, using e^{ϵ} is easier to do mathematics than using ϵ,

It can be proved that if the set of answers is discrete, Definition 2.6.1 can be equivalently stated in terms of singleton Z's [DKM+06]. Clearly, the smaller the privacy parameter ϵ is, the higher is the protection.

We shall adapt the notion of differential privacy to our framework. Apart from data points, we consider more general notions of secret information, and hence corresponding symmetric *adjacency relations* $\smile$ between secrets. This extends the dataset-based adjacency notion of "differing for only one record" to more comprehensive settings. This idea of extending the principle of differential privacy to measure the degree of protection of secrets in more general settings is also studied in [BKOB12, GHH+13, CABP13]. See also the following two examples.

Example 2.6.2 (Anonymity)**.** *In the case of anonymity the confidential data $\mathcal{U}$ are the agents' identities. Since the identities are just names without any particular structure, it is natural to assume that each name is adjacent to any other. Hence $(\mathcal{U}, \smile)$ is a clique, i.e. for all $u_1, u_2 \in \mathcal{U}$ and $u_1 \neq u_2$, we have $u_1 \smile u_2$.*

Example 2.6.3 (Geolocation)**.** *In the case of geolocation, the confidential data are the coordinates $(latitude, longitude)$ of a point on the earth's surface. If the purpose is to protect the exact location, a good definition of adjacency is: two points are adjacent if their Manhattan distance is 1, i.e. $(a_1, b_1) \smile (a_2, b_2)$ iff $|a_1 - a_2| = 1$ and $|b_1 - b_2| = 0$, or $|a_1 - a_2| = 0$ and $|b_1 - b_2| = 1$.*

Three

Modular Reasoning in a Probabilistic Process Calculus

The main goal of the present chapter is to investigate differential privacy for concurrent systems in the context of a probabilistic process calculus (CCS_p). We present a modular approach for reasoning about differential privacy, with respect to the constructs of CCS_p. More specifically, we show that the restriction, the probabilistic choice, the nondeterministic choice, and a restricted form of parallel composition are safe under composition, in the sense that they do not decrease the privacy of a system. Compositionality plays an important role in the construction and analysis of security systems: Rather than analyzing a complex system as a whole, the safe constructs allow us to split the system in parts, analyze the degree of privacy of each part separately, and combine the results to obtain the global degree of privacy.

We will use the *Crowds* protocol as an example running all through the chapter. Crowds [RR98] is an anonymity protocol which allows Internet users to perform web transactions without revealing their private identity. This is achieved by using a chain of forwarders, chosen randomly, rather than sending the message directly to the final recipient. In the standard analysis, it is often assumed that attackers stop forwarding the message after the first detection of an honest user. We provide a formal proof showing

that continuing forwarding the message after the first detection does not compromise the level of differential privacy, meaning that the assumption is indeed reasonable.

We illustrate our compositionality results by proving an anonymity-preservation property for an extension of the *Crowds* protocol. In the standard Crowds protocol, all members have the same probability of being used as forwarders, which gives the protocol a symmetric structure (cf. equations (13) and (14) in [CP06]). Unfortunately, the reality is different. Indeed, anonymity users behave as *prosumers*, i.e., the consumers of the anonymity service are at the same time its providers, as they cooperate to generate the network activity that grants anonymity to the system as a whole. Cooperation entails relaying other users' messages in order to create sufficient "doubt" as to whom the real message originator actually is. Hence, the success of such protocols depends strongly on the attitude towards cooperation with each involved individual. However, cooperation cannot be taken for granted due to the rational behavior (aka selfish behavior) of the users [NDW10, YSH12] leading to a degraded user experience. Consequently, incentive mechanisms stimulating cooperation are being increasingly used to enhance the reliability of anonymity systems. Roughly, we have two types of incentive mechanisms: *trust* based mechanisms [DBMC14, SHY10] which set up reputation systems quantifying the subjective reliance on the expected behavior of users and virtual currency which monetize the effect of prosocial behaviors [JJS13]. More recently, it has been recognized that a successful combination of both social and economic strategies should be taken into account to enhance privacy [ABLS14].

In this chapter, we consider a Crowds protocol enhanced with both a trust based mechanism and a virtual currency. The trust mechanism allows each member to establish a set of *trusted* users to whom she can forward messages. The virtual currency allows cooperating users to earn sufficient money to use the system for their own benefit. We call such "wealthy" users the *legitimate initiators* as they are the only members who can initiate transactions. This breaks the symmetry properties of the original protocol, thus making the standard analysis of [RR98] unapplicable. We argue that, in the

asymmetric case, reasoning about the protocol as a whole is difficult. But our compositional approach manages to prove an anonymity-preservation property without relying on the symmetry properties.

Furthermore, for Crowds containing trust information, we discover some cases in which the leak of information is brought in by trust information rather than the fault of the Crowds protocol. We introduce a corresponding notion of adjacency relation to rule out this factor, thus retrieving a measure of real privacy.

In [HPSE10] the authors notice that each user in the crowd must establish a path between her and a server out of a set of servers, the message forwarded in the network reveals also the identity of the target sever. This additional observation may leak additional information about the initiator of the transaction, when users' habits of Web browsing is not uniform, which is usually the case in the real world. We analyze the impact of these additional observables on the security of a protocol in the context of differential privacy, and show that compared with the work in [HPSE10], our expression is simpler, i.e. in terms of the privacy level of the channel modeling the source of the additional observation.

Summary of Contributions

- We prove that in the Crowds protocol, continuing forwarding the message after the first detection does not compromise differential privacy.

- We present a modular approach for reasoning about differential privacy for protocols expressed in a probabilistic process algebra (CCS_p).

- We apply our compositional method to prove an anonymity-preservation property for an extended version of Crowds.

- We define a notion of adjacency relation between members of Crowds involving trust information, to exclude some loss of privacy induced by the trust information.

- We show that in the setting of differential privacy, the impact of users' preference levels in Crowds has a simpler formulation than in the case

of probable innocence (which can be considered as another merit of the notion of differential privacy).

Plan of the Chapter We begin by giving background on CCS_p and differential privacy with secret labels, Crowds, and present a proposition of Crowds. Next, in Section 3.2 we investigate the compositionality of differential privacy with respect to CCS_p constructs. In Section 3.3, we apply our compositionality result to an extended Crowds. In Section 3.4 a notion of adjacency relation is tailored for Crowds with trust information. In Section 3.5, we revisit Crowds in the presence of users' preference levels. Section 3.6 and 3.7 discuss further related work and conclude.

3.1 Preliminaries

In this section we present the preliminaries for this chapter: a CCS_p with secret and observable labels introduced for the purpose of specifying information-hiding protocols, the formalism of the notion of differential privacy established therein as well as the Crowds protocol.

3.1.1 CCS_p with secret labels

In order to model security systems and protocols, we use a variant of the calculus CCS_p presented in Section 2.3. We adopt the idea in [BCP08, CPB], making a distinction between *observable* and *secret* labels.

Formally, we consider a finite set A of labels, partitioned into a set Sec of *secrets*, a set Obs of *observables* and a silent action τ. We denote labels in A, Sec and Obs by a, u and o, respectively, with primes and indices when necessary. Secret labels and the silent action τ[1] are unobservable from the point of view of outsiders (viz adversaries). For each $o \in Obs$, we assume a complementary label $\overline{o} \in Obs$ with the convention that $\overline{\overline{o}} = o$. Being unobservable, τ and secret labels have no complement.

[1]Following the tradition in concurrency theory where τ is not observable. We note, however, the compositionality result in this chapter holds independently of whether τ is observable or not.

The syntax of CCS_p with secret and observable labels is obtained from the syntax of CCS_p presented in Section 2.3 by replacing prefix $\alpha.Q$ and non-deterministic choice $Q + R$ with the following two guarded choices:

$$\begin{array}{ll} \biguplus_{i\in I} u_i.Q_i & \textit{secret choice } (u_i \in \textit{Sec}) \\ \biguplus_{i\in I} r_i.Q_i & \textit{nondeterministic choice } (r_i \in \textit{Obs} \cup \{\tau\}) \end{array}$$

Accordingly, its semantics is obtained from the semantics presented in Fig. 2.1 by replacing the rules of ACT and SUM1 with the following two rules SEC and NDC:

$$\text{SEC}\ \frac{j \in I}{\biguplus_{i\in I} u_i.Q_i \xrightarrow{u_j} \delta(Q_j)} \qquad \text{NDC}\ \frac{j \in I}{\biguplus_{i\in I} r_i.Q_i \xrightarrow{r_j} \delta(Q_j)}$$

We need to adjust the definition of scheduler for the distinction between secret and observable labels. Following [BCP08, CPB] we assume secret labels to be the *inputs* of the system. Secrets are given as input to a scheduler and determine completely the secret choices, namely a secret label can only be performed if it matches the input. The scheduler then has to resolve the residual nondeterminism, which is originated by nondeterministic choice and parallel operator. From an outsider's point of view, only observable labels can be seen.

The definition of a scheduler with secret labels is specified as follows. Let $\alpha_{|Sec}$ and $\alpha_{|Obs}$ be the projection of α on Sec and Obs, respectively. For instance if $\alpha = Q_1u_1Q_2u_2Q_3\tau Q_4u_3Q_5o_1Q_6$ where for all i, $u_i \in Sec$ and $o_i \in Obs$, then $\alpha_{|Sec} = u_1u_2u_3$, $\alpha_{|Obs} = o_1$. The projection of an execution on secret labels should be consistent with a given input.

Definition 3.1.1 (Scheduler with secret labels)**.** *Let Q be a process in CCS_p and $\mathcal{A}$ be the probabilistic automaton generated by Q. A scheduler is a function $\zeta : Sec^* \times Exec^*(\mathcal{A}) \rightharpoonup D$ such that for a given secret $\vec{u} = u_1u_2\cdots u_n$ and an execution α, $\alpha_{|Sec} = u_1u_2\cdots u_m$ with $m \leq n$ iff $\zeta(\vec{u})(\alpha)$ is defined. Furthermore, let $\zeta(\vec{u})(\alpha) = (lstate(\alpha), a, \mu)$, if $m < n$ and $a \in Sec$ then $a = u_{m+1}$, else if $m = n$ then $a \notin Sec$. We will write $\zeta_{\vec{u}}(\alpha)$ for $\zeta(\vec{u})(\alpha)$.*

We now define the *execution tree* of a CCS_p term with secret labels, in a way similar to what is done in probabilistic automata. The main difference

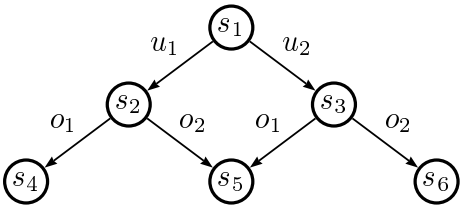

Figure 3.1: A process with secret labels.

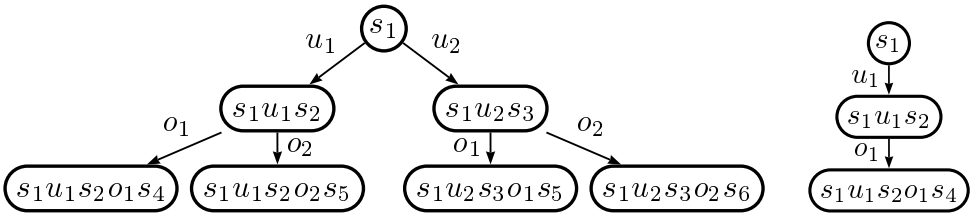

Figure 3.2: An execution tree with secret labels.

is that in our case the execution tree depends not only on the scheduler, but also on the secret input.

Definition 3.1.2 (Execution tree with secret labels)**.** *Let $\mathcal{A} = (S, Q, A, D)$ be the probabilistic automaton generated by Q. Given an input $\vec{u}$ and a scheduler ζ, the* execution tree *of Q, denoted by $etree(Q, \vec{u}, \zeta)$, is a fully probabilistic automaton $\mathcal{A}' = (S', Q, A, D')$ such that:*

(i) $S' = Exec(\mathcal{A})$,

(ii) $(\alpha, a, \mu') \in D'$ *iff* $\zeta_{\vec{u}}(\alpha) = (lstate(\alpha), a, \mu)$ *for some* μ *and* $\mu'(\alpha a Q) = \mu(Q)$

Example 3.1.3. *Figure 3.1 shows a probabilistic automaton of a process with secret labels, where $Sec = \{u_1, u_2\}$ and $Obs = \{o_1, o_2\}$. Its execution tree is shown in the left part of Figure 3.2. Define a scheduler $\zeta_{u_1}(s_1) = (s_1, u_1, \Delta(s_2))$, $\zeta_{u_1}(s_1u_1s_2) = (s_2, o_1, \Delta(s_4))$ and undefined on the remaining execution states, we obtain a fully probabilistic automaton as shown in the right part of Figure 3.2.*

3.1.2 Process terms as channels

We now show how CCS_p terms can be used to specify systems manipulating confidential information.

A system can be seen as an information-theoretic channel [CT91]. A finite sequence of secret labels constitutes the *secret information* (or *a secret*), given as an input to the channel, and a finite sequence of observable labels constitutes the *public information* (or *an observable*), obtained as an output from the channel. Given an input $\vec{u}$, a run of the system will produce an output $\vec{o}$ with a certain probability which depends on the input, on the randomized operations performed by the system and also on the scheduler ζ resolving nondeterminism. We denote the probability by $p_\zeta(\vec{o}|\vec{u})$. Given a scheduler ζ, the probabilities $p_\zeta(\vec{o}|\vec{u})$ constitute a matrix M_ζ, which is called the *channel matrix*, where the rows are indexed by the elements of Sec^* and the columns are indexed by the elements of Obs^*.(See some toy examples of channel matrices in Example 3.1.6 and 3.1.7.) We present a formal definition of channel matrix below.

First we define a probability measure on executions. Given an input $\vec{u} \in Sec^*$ and a scheduler ζ, the execution tree $\mathcal{A}' = etree(Q, \vec{u}, \zeta)$ induces a probability measure over executions as follows. The basic measurable events are the cones of finite executions, where the cone of a finite execution α, denoted by C_α, is the set $\{\alpha' \in Exec^*(\mathcal{A}') \cup Exec(\mathcal{A}') | \alpha \leq \alpha'\}$, where $\leq$ is the standard prefix preorder on sequences. The probability of a cone C_α, denoted by $p_\zeta(\alpha|\vec{u})$, is simply the multiplication of the probabilities along the execution. More precisely, let $\alpha = Q_{init} a_0 Q_1 a_1 Q_2 \cdots a_n Q_n$ and α_i be the prefix of α up to the state Q_i, where for each i, $\zeta_{\vec{u}}(\alpha_i) = (Q_i, a_i, \mu_i)$, then $p_\zeta(\alpha|\vec{u}) = \prod_{i=1}^{n} \mu_i(Q_{i+1})$.

Now we are ready to give the formal definition of a channel matrix.

Definition 3.1.4 (Channel matrix)**.** *Given a process term Q and a scheduler ζ, the channel matrix $M_\zeta(Q)$ is defined as the matrix such that, for each row $\vec{u} \in Sec^*$ and column $\vec{o} \in Obs^*$, $p_\zeta(\vec{o}|\vec{u})$ is the probability of the set of finite executions in $etree(Q, \vec{u}, \zeta)$ whose projection in Obs is $\vec{o}$, i.e. $p_\zeta(\vec{o}|\vec{u}) = \sum_{\alpha_{|Obs} = \vec{o}} p_\zeta(\alpha|\vec{u})$.*

3.1.3 Differential Privacy in CCS$_p$ with secret labels

In Section 2.6 we have introduced the basic notion of differential privacy. Here we adapt it to measure the degree of privacy provided by a CCS$_p$ process with secret labels. Let $\mathcal{U} \subseteq Sec^*$, $\mathcal{O} \subseteq Obs^*$.

Definition 3.1.5 (Differential Privacy in CCS$_p$ with secret labels)**.** *A process Q provides* ϵ-differential privacy *(ϵ-DP) iff for all schedulers ζ, for all secret inputs $\vec{u}_1, \vec{u}_2 \in \mathcal{U}$ such that $\vec{u}_1 \backsim \vec{u}_2$, and for all observable $\vec{o} \in \mathcal{O}$,*

$$p_\zeta(\vec{o}|\vec{u}_1) \leq e^\epsilon \, p_\zeta(\vec{o}|\vec{u}_2)$$

We use $dp_\zeta[\![Q]\!]$ to denote the smallest value ϵ of differential privacy that Q enjoys under the scheduler ζ. Furthermore we define

$$dp[\![Q]\!] = \sup_\zeta dp_\zeta[\![Q]\!]$$

In this chapter, we consider a simple model of attacker, in which attacker can only interact with the system through observable labels. In the above definition, we take the worst case over all schedulers, which is the typical way of resolving a class of nondeterminism. In the real world, the nondeterminism can be considered introduced by an adversary interplaying with the system trying to learn as much information as possible about $\vec{u}$.

Note that if there are both zero and non-zero probabilities occurring in the same column of the channel matrix, when the respective secrets are connected by $\backsim$, then the process does not provide differential privacy for any ϵ. We give some simple examples to illustrate the above definition. We denoted by $\Delta(Q)$ the set of *all schedulers* for a process Q.

Example 3.1.6. *Let $Sec = \{u_1, u_2\}$, $u_1 \backsim u_2$ and $Obs = \{o_1, o_2, a\}$, and consider the following processes: $Q_1 = o_1.\mathbf{0} \oplus_{0.3} o_2.\mathbf{0}$, $Q_2 = o_1.\mathbf{0} \oplus_{0.5} o_2.\mathbf{0}$, $Q_3 = o_1.\mathbf{0} \oplus_{0.8} o_2.\mathbf{0}$, $Q = a.Q_1 \boxplus a.Q_2 \boxplus a.Q_3$ and $Q' = u_1.Q \boxplus u_2.Q$.*

The probabilistic automaton of Q' is shown in the left part of Figure 3.3.

For the process Q_1, we have $\Delta(Q_1) = \{\varnothing\}$ and $M(Q_1) =$

o_1	o_2
0.3	0.7

.

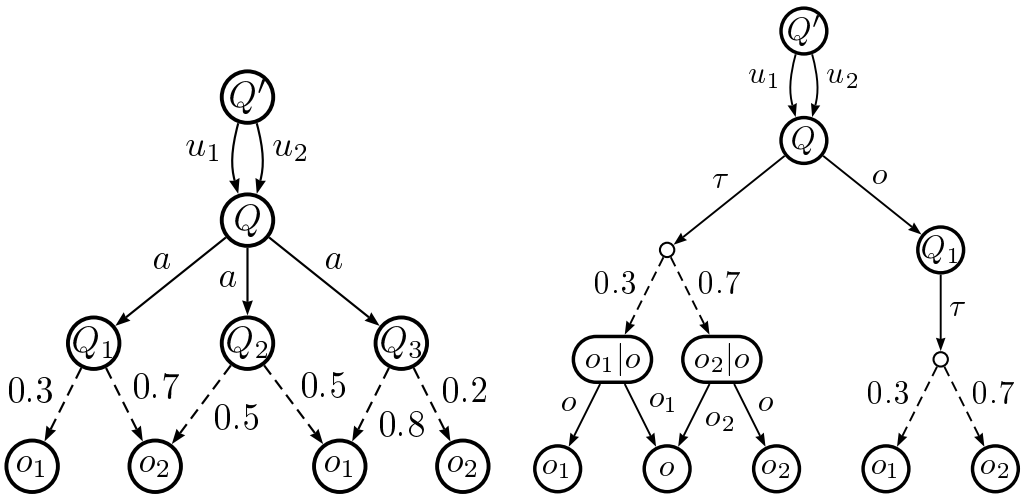

Figure 3.3: The probabilistic automata of Example 3.1.6 and 3.1.7.

For the process Q_2, we have $\Delta(Q_2) = \{\varnothing\}$ and $M(Q_2) =$

o_1	o_2
0.5	*0.5*

.

For the process Q_3, we have $\Delta(Q_3) = \{\varnothing\}$ and $M(Q_3) =$

o_1	o_2
0.8	*0.2*

.

For the process Q, we have $\Delta(Q) = \{\zeta_1, \zeta_2, \zeta_3\}$ with ζ_i selecting Q_i.

For the process Q' we can define the scheduler ζ' which selects ζ_1 and ζ_3 for the secret u_1 and u_2, respectively. The corresponding matrix is $M_{\zeta'}(Q') =$

	ao_1	ao_2
u_1	*0.3*	*0.7*
u_2	*0.8*	*0.2*

, which gives $(\ln 3.5)$-differential privacy.

Example 3.1.7. *Let $Sec = \{u_1, u_2\}$, $u_1 \backsim u_2$ and $Obs = \{\tau, o, o_1, o_2\}$, and consider the processes $Q_1 = o_1.\mathbf{0} \oplus_{0.3} o_2.\mathbf{0}$, $Q_2 = o.\mathbf{0}$, $Q = Q_1 \,|\, Q_2$, and $Q' = u_1.Q \boxplus u_2.Q$.*

The probabilistic automaton of Q' is shown in the right part of Figure 3.3. Through the steps similar to the above example, we can find a scheduler that if the secret is u_1, first performs a label in Q_1 and then Q_2, if the secret is u_2, first selects Q_2 and then Q_1, thus producing a matrix

breaking differential privacy, as follows:

	o_1o	o_2o	oo_1	oo_2
u_1	*0.3*	*0.7*	*0*	*0*
u_2	*0*	*0*	*0.3*	*0.7*

.

The supremum probability in the definition of DP is actually a maximum. First we shall define a suitable metric on the set $\Delta(Q)$ of schedulers of a process Q.

Definition 3.1.8. *Consider a CCS_p process Q, let $\mathcal{A}$ be the probabilistic automaton generated by Q. We define a distance d between schedulers in $\Delta(Q)$ as follows:*

$$d(\zeta,\zeta') = \begin{cases} 2^{-m} & \text{if } m = \min\{|\alpha| \mid \vec{u} \in \mathcal{U}, \alpha \in Exec^*(\mathcal{A}) \text{ and } \zeta_{\vec{u}}(\alpha) \neq \zeta'_{\vec{u}}(\alpha)\} \\ 0 & \text{if } \zeta_{\vec{u}}(\alpha) = \zeta'_{\vec{u}}(\alpha) \text{ for all } \vec{u} \in \mathcal{U}, \alpha \in Exec^*(\mathcal{A}) \end{cases}$$

where $|\alpha|$ represents the length of α.

Note that $\mathcal{A}$ is finitely branching, both in the nondeterministic and in the probabilistic choices. Hence we have the following (standard) result:

Proposition 3.1.9. $(\Delta(Q), d)$ *is a* sequentially compact *metric space, i.e., every sequence has a subsequence that converges to a limit in $\Delta(Q)$.*

In the following proposition, we show that there exists a scheduler that gives the maximum probability of DP.

Proposition 3.1.10. *For every process Q we have*

$$dp[\![Q]\!] = \sup_{\zeta} dp_{\zeta}[\![Q]\!] = \max_{\zeta} dp_{\zeta}[\![Q]\!]$$

Proof. If there exists a scheduler $\zeta \in \Delta(Q)$ such that for two inputs $\vec{u}_1 \backsim \vec{u}_2$ and an output $\vec{o} \in \mathcal{O}$, $p_\zeta(\vec{o}|\vec{u}_1) = 0$ and $p_\zeta(\vec{o}|\vec{u}_2) \neq 0$, then trivally $dp_\zeta[\![Q]\!] = \infty$, the scheduler ζ achieves the supremum value of $dp[\![Q]\!]$.

Thus in the following, we consider the case where for any scheduler $\zeta \in \Delta(Q)$, any $\vec{u}_1 \backsim \vec{u}_2$ and $\vec{o} \in \mathcal{O}$, $p_\zeta(\vec{o}|\vec{u}_1) = 0$ iff $p_\zeta(\vec{o}|\vec{u}_2) = 0$. By Def. 3.1.5,

$$dp_\zeta[\![Q]\!] = \max_{\vec{u}_1 \backsim \vec{u}_2} \max_{\vec{o} \in \mathcal{O}} \Big| \ln \frac{p_\zeta(\vec{o}|\vec{u}_1)}{p_\zeta(\vec{o}|\vec{u}_2)} \Big|$$

$dp_\zeta[\![Q]\!]$ is a continuous function from $(\Delta(Q), d)$ to $([0, +\infty), d')$, where d' is the standard distance on real numbers. By Prop. 3.1.9, $(\Delta(Q), d)$ is sequentially compact. Consequently, $(\{dp_\zeta[\![Q]\!] \mid \zeta \in \Delta(Q)\}, d')$ is also sequentially compact. Let $\{\zeta_n\}_n$ be a sequence such that for any n,

$$|dp_{\zeta_n}[\![Q]\!] - \sup_\zeta dp_\zeta[\![Q]\!]| \leq 2^{-n}$$

We have that $\{dp_{\zeta_n}[\![Q]\!]\}_n$ is convergent and

$$\lim_n dp_{\zeta_n}[\![Q]\!] = \sup_\zeta dp_\zeta[\![Q]\!]$$

Consider now a convergent subsequence $\{\zeta_{n_j}\}_j$ of $\{\zeta_n\}_n$. By continuity of $dp_\zeta[\![Q]\!]$, we have

$$\lim_n dp_{\zeta_n}[\![Q]\!] = \lim_j dp_{\zeta_{n_j}}[\![Q]\!] = dp_{\lim_j \zeta_{n_j}}[\![Q]\!]$$

which concludes the proof. □

3.1.4 The Crowds protocol

Here we recall the Crowds protocol in details and then present the formal proof of the fact that continuing forwarding messages after the first detection gains no more information for attackers.

Crowds is an anonymity protocol which allows users to send messages without revealing their identity. A crowd is a group of n participants constituted by m *honest members* and c $(= n - m)$ *corrupted members* (the *attackers*). The destination of messages is named the *server*. An example of crowds is shown in Fig. 3.4. The protocol works as follows:

- When a member, called the *initiator*, wants to send a message to the server, instead of sending it directly to the server, she randomly selects a member in the crowd and she forwards the message to this member.

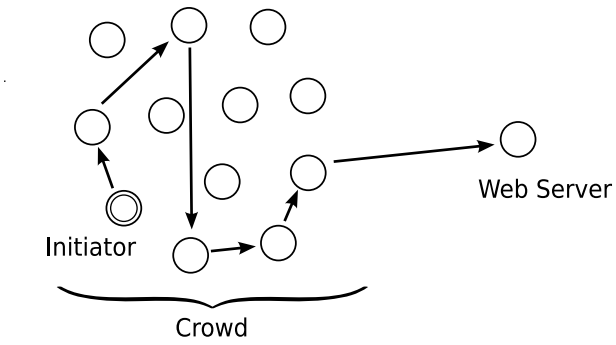

Figure 3.4: The Crowds protocol

- Every member who receives the message, either

  - with probability $1 - p_f$, delivers the message to the server, or
  - with probability p_f randomly selects another member (possibly herself) in the crowd as the new *forwarder* and relays the message to this new forwarder to repeat the same procedure again.

In this way, even if the message is caught by an attacker, the attacker cannot be sure whether the previous forwarder is the initiator or just a forwarder on behalf of somebody else. Members (including attackers) are assumed to have only access to messages routed through them, so that they only know the identities of their immediate predecessors and successors in the path, and of the destination server. For simplicity we assume that once an attacker receives a message from an honest member, it will terminate after reporting the detection. The reason is that by forwarding the message after the first detection, attackers can not gain more useful information. Its formal proof is illustrated below in Proposition 3.1.11.

We use H and B to denote the set of honest members and of attackers, respectively. $C = H \cup B$ represents the set of identities of all participants in the crowd. Let $C = \{1, 2, \cdots, n\}$. We denote by u_i where $i \in H$ the event that user i is the initiator, by o_j where $j \in H$ the event that user j is detected by some attacker. We denote by o_j^k where $k \in B, j \in H$ the event that user j gets detected by the attacker k. Namely, it represents the event that an honest member j unluckily forwards a message to an attack k. The

honest member's identity is exposed to the attack, so this event can also be considered observable. We denote by $\overline{NO}$ that there is an attacker in the path, and $\overline{OK}$ that the request is successfully sent to the server without revealing the identity of any user. The probability that user j is detected, given that user i is the initiator and the request is indeed forwarded to a corrupted member, is denoted by $p(o_j|u_i, \overline{NO})$. For standard Crowds, Reiter and Rubin proved in [RR98] that the following holds:

$$p(o_j|u_i, \overline{NO}) = \begin{cases} 1 - \frac{m-1}{n} p_f & i = j \\ \frac{1}{n} p_f & i \neq j \end{cases} \tag{3.1}$$

We present below the formal proof of the fact that by forwarding the message after the first detection, attackers can not gain more useful information. More precisely, Crowds in which attackers continue forwarding the message after detections provides the same level of privacy as Crowds in which attackers terminate after reporting the first detection.

Proposition 3.1.11. *Let Crowds represent a crowd in which the attackers terminate after the first detection, while FCrowds represent the same crowd but in which the attackers continue to forward the message after detections. We have*

$$dp[\![Crowds]\!] = dp[\![FCrowds]\!]$$

Proof. In *Crowds*, only one user's identity is observed. In *FCrowds*, a sequence of users' identities is observed. We denote the sequence of detections by $o_j^i\alpha$, in which o_j^i represents that user j is the first detected member and caught by attacker i, α represents the rest of the sequence. We show that in the channel matrices of *Crowds* and *FCrowds*, given two arbitrary initiators u and u', the following ratios turn out to be same,

$$\frac{p(o_j^i|u)}{p(o_j^i|u')} = \frac{p(o_j^i\alpha|u)}{p(o_j^i\alpha|u')}$$

Indeed, since once the first detection o_j^i is observed, the identity of the initiator u does not add any additional information about the following

detections α, and the behavior of attacker i does not depend on u, thus α and u are conditionally independent given o^i_j. We have:

$$\begin{aligned} p(o^i_j\alpha|u) &= p(\alpha|u,o^i_j)p(o^i_j|u) && \text{(by Bayes' law)} \\ &= p(\alpha|o^i_j)p(o^i_j|u) && \text{(the conditional independence} \\ & && \text{between } \alpha \text{ and } u \text{ given } o^i_j) \end{aligned}$$

So the required equality holds:

$$\frac{p(o^i_j\alpha|u)}{p(o^i_j\alpha|u')} = \frac{p(\alpha|o^i_j)p(o^i_j|u)}{p(\alpha|o^i_j)p(o^i_j|u')} = \frac{p(o^i_j|u)}{p(o^i_j|u')}$$

It is easy to see that the channel matrix of *FCrowds* is a fine and expanded one of *Crowds*, but the ratios between the rows are not changed. Hence, according to the definition of differential privacy (Def. 3.1.5), *Crowds* and *FCrowds* have the same privacy level. □

3.1.5 Relation between differential privacy and anonymity

We will use Crowds as a case study to illustrate our framework for privacy. The purpose of Crowds is anonymity, i.e. to conceal the initiator's identity. We show that there is a close relation between differential privacy and some existing notions of anonymity.

An *anonymity system* is a special private system in which the secret information to be concealed is a set of identities of anonymous users. Consider an anonymity system Q with m anonymous users. Let the set of secrets $\mathcal{U} = \{u_1, u_2, \ldots, u_m\}$, and the set of observables $\mathcal{O} = \{o_1, o_2, \ldots, o_m\}$. We denote by $p_\zeta(o_j|u_i)$ the probability of observing user j, given that the scheduler is ζ and the anonymous initiator is user i. We omit ζ when Q is a purely probabilistic system, i.e., without non-deterministic behavior. Note that the identities $(\mathcal{U}, \backsim)$ form a clique (cf. Example 2.6.2).

Strong anonymity

Strong anonymity for purely probabilistic systems was formalized by Chaum [Cha88] as the property that the observation of user k does not change the probabilistic knowledge of the culprit's identity i, i.e. $p(u_i|o_k) = p(u_i)$ for every k and i. Bhargava and Palamidessi [BP05] extended this notion to probabilistic and non-deterministic systems, essentially by requiring that the equation holds under any scheduler, and showed that it is equivalent to the equality of $p_\zeta(o_k|u_i) = p_\zeta(o_k|u_j)$ for every k, i, j and ζ. The next proposition is an immediate consequence of this alternative characterization.

Proposition 3.1.12. *An anonymity system Q is strongly anonymous if $dp[\![Q]\!] = 0$.*

Proof.

$dp[\![Q]\!] = 0$

$\Rightarrow$ Under any scheduler ζ, for every k, i and j, $\dfrac{p_\zeta(o_k|u_i)}{p_\zeta(o_k|u_j)} \leq e^0 = 1$.

(by Definition 3.1.5)

$\Rightarrow$ Under any scheduler ζ, for every k, i and j, $p_\zeta(o_k|u_i) = p_\zeta(o_k|u_j)$.

$\Rightarrow$ Q is strongly anonymous. (by definition in [BP05])

□

Probable innocence

Probable innocence for purely probabilistic systems was defined in [RR98] as the property that, to the eyes of an observer, each user is more likely to be innocent rather than culpable (of having initiated the message). In [CP06] it was shown that this is equivalent to requiring $(m-1)p(o_k|u_i) \geq p(o_k|u_j)$ for all k, i and j, where m is the number of anonymous users. (Note that in Crowds the number of anonymous users is not the number of all users, but the number of honest users.) The next Proposition follows immediately from this characterization.

Proposition 3.1.13. *A purely probabilistic anonymity system Q has probable innocence if $dp[\![Q]\!] \leq \ln(m-1)$, where m is the number of anonymous users.*

Proof.

$$
\begin{aligned}
& dp[\![Q]\!] \leq \ln(m-1) \\
\Rightarrow \quad & \text{For all } k, i \text{ and } j, \frac{p(o_k|u_i)}{p(o_k|u_j)} \leq m-1. && \text{(Def. 3.1.5)} \\
\Rightarrow \quad & \text{For all } k, i \text{ and } j, (m-1)p(o_k|u_j) \geq p(o_k|u_i). \\
\Rightarrow \quad & Q \text{ has probable innocence.} && \text{(Definition in [CP06])}
\end{aligned}
$$

□

3.2 Modular Reasoning

In this section we investigate the compositional properties of CCS_p constructs with respect to differential privacy and state the first important result of the thesis. We start by introducing a notion called *safe component*.

Definition 3.2.1. *Consider a process Q, and observables $o_1, o_2, \ldots, o_k$, we say that $(\nu o_1, o_2, \ldots, o_k)Q$ is a* safe component *if*

(i) Q does not contain any secret label, and

(ii) all the observable labels of Q are included in $o_1, o_2, \ldots, o_k$.

It is easy to see that all observable labels are restricted in a safe component, nothing can be observed, thus its privacy level is 0.

We now show that non-deterministic choice, probabilistic choice, the restriction operator and a restricted form of parallel composition are *safe*, in the sense that combining components under these operators does not compromise the privacy of the system.

Theorem 3.2.2. *Let I be a finite (index) set and $\{Q_i\}_{i\in I}$ be a family of processes s.t. for each i, $dp[\![Q_i]\!] = \epsilon_i$. Then:*

(1) $dp[\![\biguplus_i o_i.Q_i]\!] \leq \max_i\{\epsilon_i\}$;

(2) $dp[\![\bigoplus_i p_i Q_i]\!] \leq \max_i\{\epsilon_i\}$;

(3) $dp[\![(\nu o)Q_1]\!] \leq \epsilon_1$;

(4) Assume that $(\nu o_1, o_2, \ldots, o_k)Q_1$ *is a safe component, that* Q_1 *and* Q_2 *can communicate with each other only via the labels of the set* $\{o_h, \ldots, o_k\}$, *with* $1 \leq h \leq k$, *and that* $dp[\![(\nu o_1, \ldots, o_{h-1})Q_2)]\!] = \epsilon_2$. *Then*

$$dp[\![(\nu o_1, o_2, \ldots, o_k)\,(Q_1 \mid Q_2)]\!] \leq \epsilon_2$$

Proof. 1. Let $Q = \biguplus_i o_i.Q_i$. Consider an arbitrary scheduler ζ of Q, by Rule NDC, ζ resolves the top non-deterministic choice by arbitrarily choosing a label $o_j, j \in \{1, 2, \ldots, h\}$. Define a scheduler ζ_j as ζ except for the removal of the first state and the first step o_j from the execution fragments in the domain. Obviously, the scheduler ζ_j is compatible with process term Q_j, namely, ζ_j is one of Q_j's schedulers. For every conditional probability $p_\zeta(o_j\vec{o}|\vec{u})$ in $M_\zeta(Q)$, there exists a one to one corresponding element $p_{\zeta_j}(\vec{o}|\vec{u})$ in $M_{\zeta_j}(Q_j)$, such that

$$p_\zeta(o_j\vec{o}|\vec{u}) = p_{\zeta_j}(\vec{o}|\vec{u}).$$

From the level of differential privacy that Q_j gives, we derive

$$dp_\zeta[\![Q]\!] = dp_{\zeta_j}[\![Q_j]\!] \leq \epsilon_j \leq \max_i\{\epsilon_i\}$$

which concludes the proof in this case.

2. Let $Q = \bigoplus_i p_i Q_i$. Consider an arbitrary scheduler ζ for Q. After one subprocess Q_i is randomly chosen according to its probability, ζ must be compatible with one scheduler ζ_i of this subprocess, resolving all the nondeterminism in it. It holds that the conditional probability $p_\zeta(\vec{o}|\vec{u})$ in $M_\zeta(Q)$ and the corresponding one $p_{\zeta_i}(\vec{o}|\vec{u})$ in each matrix $M_{\zeta_i}(Q_i)$ satisfy the following relation.

$$p_\zeta(\vec{o}|\vec{u}) = \sum_i p_i \cdot p_{\zeta_i}(\vec{o}|\vec{u})$$

It is easy to see that for all secret inputs $\vec{u}$, $\vec{u}'$ such that $\vec{u} \backsim \vec{u}'$,

$$\begin{aligned}
\frac{p_\zeta(\vec{o}|\vec{u})}{p_\zeta(\vec{o}|\vec{u}')} &= \frac{\sum_i p_i \cdot p_{\zeta_i}(\vec{o}|\vec{u})}{\sum_i p_i \cdot p_{\zeta_i}(\vec{o}|\vec{u}')} \\
&\leq \frac{\sum_i p_i \cdot e^{\epsilon_i} p_{\zeta_i}(\vec{o}|\vec{u}')}{\sum_i p_i \cdot p^i(\vec{o}|\vec{u}')} \quad \text{(since } M_\zeta(Q_i) \text{ gives } \epsilon_i\text{-d.p.)} \\
&\leq e^{\max_i\{\epsilon_i\}} \frac{\sum_i p_i \cdot p_{\zeta_i}(\vec{o}|\vec{u}')}{\sum_i p_i \cdot p_{\zeta_i}(\vec{o}|\vec{u}')} \\
&= e^{\max_i\{\epsilon_i\}}
\end{aligned}$$

which concludes the proof in this case.

3. Let $Q = (\nu o)Q_1$. By the rule RES, Q is not able to perform the label o. Its execution tree $etree(Q)$ can be obtained from $etree(Q_1)$ by cutting all transitions labelled by o and removing the whole subtrees following those $\xrightarrow{o}$ transitions. From the point of view of channel matrices, the columns which have become the same after the aforementioned removal are collapsed into one and the same column. We show that this transformation does not increase the level of DP.

   Formally, let $\mathcal{O}'$ denote the set of observables of Q. Consider an arbitrary scheduler ζ' of Q. Observe that there exists a scheduler ζ of Q_1, such that, for every conditional probability $p_{\zeta'}(\vec{o}'|\vec{u})$ (where $\vec{o}' \in \mathcal{O}'$) in $M_{\zeta'}(Q)$, the following equation holds.

$$p_{\zeta'}(\vec{o}'|\vec{u}) = \sum_{f(\vec{o})=\vec{o}'} p_\zeta(\vec{o}|\vec{u})$$

   where $p_\zeta(\vec{o}|\vec{u})$ denotes the conditional probability in $M_\zeta(Q_1)$, and the function $f(\vec{o})$ cuts the sub-trace following the label o. More precisely,

let $\vec{o} = o_1, o_2, \ldots, o_k$,

$$f(\vec{o}) = \begin{cases} \vec{o} & \text{if the label } o \text{ does not occur in the sequence } \vec{o} \\ o_1, o_2, \ldots, o_i & \text{if the label } o \text{ first occurs in the sequence } \vec{o} \text{ at the position } i+1 \text{ with } i < k. \end{cases} \tag{3.2}$$

It's easy to get that for all secret inputs $\vec{u}_1, \vec{u}_2 \in \mathcal{U}$ such that $\vec{u}_1 \backsim \vec{u}_2$, and for all observable $\vec{o}' \in \mathcal{O}'$,

$$\frac{p_{\zeta'}(\vec{o}'|\vec{u}_1)}{p_{\zeta'}(\vec{o}'|\vec{u}_2)} = \frac{\sum_{f(\vec{o})=\vec{o}'} p_\zeta(\vec{o}|\vec{u}_1)}{\sum_{f(\vec{o})=\vec{o}'} p_\zeta(\vec{o}|\vec{u}_2)} \leq \frac{\sum_{f(\vec{o})=\vec{o}'} e^{\epsilon_1} p_\zeta(\vec{o}|\vec{u}_2)}{\sum_{f(\vec{o})=\vec{o}'} p_\zeta(\vec{o}|\vec{u}_2)} = e^{\epsilon_1}$$

which shows that $M_{\zeta'}(Q)$ enjoys ϵ_1-differential privacy.

4. Let $Q = (\nu o_1, o_2, \ldots, o_k)\,(Q_1 \mid Q_2)$. The proof proceeds by reducing the execution tree of Q under an arbitrary scheduler ζ to a new one $etree'(Q, \zeta)$. This new tree enjoys a level of differential privacy which is at most as safe as the one of the original $etree(Q, \zeta)$, while it is isomorphic to the execution tree of $(\nu o_1, \ldots, o_{h-1})Q_2$ under a certain scheduler ζ_2. We derive,

   $$dp[\![etree(Q, \zeta)]\!] \leq dp[\![etree'(Q, \zeta)]\!] = dp[\![etree((\nu o_1, \ldots, o_{h-1})Q_2, \zeta_2)]\!] \leq \epsilon_2$$

   which proves that the process Q enjoys ϵ_2-differential privacy.

   The reduction from $etree(Q, \zeta)$ to $etree'(Q, \zeta)$ is described as follows. First we give the definitions of *Q_1's positions* and *Q_2's positions*. Consider an arbitrary state α in $etree(Q, \zeta)$, and let $(\nu o_1, o_2, \ldots, o_k)\,(Q_1' \mid Q_2')$ be the generic process term of $lstate(\alpha)$. From the assumption that $(\nu o_1, o_2, \ldots, o_k)Q_1$ is a safe component, there are three possible kinds of transitions performable from the state according to the operational semantics.

   (a-step) $(\nu o_1, o_2, \ldots, o_k)\,(Q_1' \mid Q_2') \xrightarrow{a} (\nu o_1, o_2, \ldots, o_k)\,(\mu \mid Q_2')$ due to a transition $Q_1' \xrightarrow{a} \mu$. In this case, a must be τ, be-

cause Q_1 does not contain secret labels and all its observable labels are included in $\{o_1, o_2, \ldots, o_k\}$. Assume that $\mu = \sum_i p_i\, Q'_{1i}$, where $Q'_{1i} \in \mathit{supp}(\mu)$. Then we have the distribution $(\nu o_1, o_2, \ldots, o_k)\,(\mu \mid Q'_2) = \sum_i p_i\, (\nu o_1, o_2, \ldots, o_k)\,(Q'_{1i} \mid Q'_2)$.

(b-step) $(\nu o_1, o_2, \ldots, o_k)\,(Q'_1 \mid Q'_2) \xrightarrow{a} (\nu o_1, o_2, \ldots, o_k)\,(Q'_1 \mid \mu)$ due to a transition $Q'_2 \xrightarrow{a} \mu$, with a not included in $\{o_1, o_2, \ldots, o_k\}$.

(c-step) $(\nu o_1, o_2, \ldots, o_k)\,(Q'_1 \mid Q'_2) \xrightarrow{\tau} (\nu o_1, o_2, \ldots, o_k)\,\delta(Q''_1 \mid Q''_2)$ due to the transitions $Q'_1 \xrightarrow{a} \delta(Q''_1)$ and $Q'_2 \xrightarrow{\bar{a}} \delta(Q''_2)$. As assumed in the condition, a must be an observable in $\{o_h, o_{h+1}, \ldots, o_k\}$.

We define Q_1*'s positions* (resp. Q_2*'s positions*) as the set of states in $\mathit{etree}(Q, \zeta)$ where ζ chooses a transition of type (a) (resp. a transition of type (b) or (c)). Recall that the execution α is an arbitrary state in $\mathit{etree}(Q, \zeta)$, the tree $\mathit{etree}'(Q, \zeta)$ is obtained by replacing each Q_1's position with its subtree $\mathit{etree}(\alpha\tau(\nu o_1, o_2, \ldots, o_k)\,(Q'_{1m} \mid Q'_2), \zeta)$ which gives the maximal value of differential privacy among all its subtrees, that is,

$$m = \operatorname*{argmax}_i \mathit{dp}[\![\mathit{etree}(\alpha\tau(\nu o_1, o_2, \ldots, o_k)\,(Q'_{1i} \mid Q'_2), \zeta)]\!]$$

By simple induction on the depth of the tree, we obtain that $\mathit{dp}[\![\mathit{etree}(Q, \zeta)]\!]$ is an increasing function of its subtrees. Then by the previous result about probabilistic choice, we have

$$\mathit{dp}[\![\mathit{etree}(Q, \zeta)]\!] \leq \mathit{dp}[\![\mathit{etree}'(Q, \zeta)]\!]$$

Note that after the process Q_1's impact on safety is resolved, all states left in $\mathit{etree}'(Q, \zeta)$ are Q_2's positions. It is easy to find a corresponding scheduler ζ_2 for the execution tree of $(\nu o_1, \ldots, o_{h-1})Q_2$ such that

- for every b-step in $\mathit{etree}'(Q, \zeta)$, ζ_2 chooses the same transition in $\mathit{etree}((\nu o_1, \ldots, o_{h-1})Q_2, \zeta_2)$, i.e. $Q'_2 \xrightarrow{a} \mu$ with $a \notin \{o_1, o_2, \ldots, o_k\}$,
- for every c-step in $\mathit{etree}'(Q, \zeta)$, ζ_2 chooses the same transition in $\mathit{etree}((\nu o_1, \ldots, o_{h-1})Q_2, \zeta_2)$, i.e. $Q'_2 \xrightarrow{\bar{a}} \delta(Q''_2)$ with $a \in \{o_h, o_{h+1}, \ldots, o_k\}$.

Observe now that $etree'(Q, \zeta)$ is isomorphic to $etree((\nu o_1, \ldots, o_{h-1})Q_2, \zeta_2)$, which concludes the proof in this case.

□

Properties (1) and (2) point out that the degree of privacy of a system, consisting of some subsystems in a non-deterministic or probabilistic choice, is determined by the subsystem with the lowest degree of privacy. Properties (3) and (4) intuitively say that, turning an observable label to be unobservable, or paralleling with a safe component, maintain the level of privacy.

Unfortunately secret choice and the unrestricted form of parallel composition do not preserve privacy, essentially due to the presence of nondeterminism. The replication operator is like an unrestricted parallel composition, and therefore it does not preserve privacy either. These are illustrated by the following counterexamples, some of which are taken from [BCP08]. (In Examples 3.2.3 - 3.2.6, we use the original definition of the adjacency relation, that is, the difference in only one label.)

Example 3.2.3 (Secret choice does not preserve privacy)**.** *Let* $Sec = \{u_1, u_2\}$. *Consider the process* $Q = o_1.\mathbf{0} \boxplus o_2.\mathbf{0}$. *Clearly,* Q *provides* 0*-differential privacy, because for every sequence* $\vec{u} \in \mathcal{U}$ *we have* $p(o_1|\vec{u}) = p(o_2|\vec{u})$. *Consider now a new process* $Q' = u_1.Q \boxplus u_2.Q$, *and the scheduler* ζ *for* Q' *which selects* o_1 *if the secret is* u_1, *and* o_2 *if the secret is* u_2. *The resulting matrix under* ζ *does not preserve differential privacy, since* $p(o_1|u_1\vec{u}) = p(o_2|u_2\vec{u}) = 1$ *while* $p(o_1|u_2\vec{u}) = p(o_2|u_1\vec{u}) = 0$.

Example 3.2.4 (The need of condition (i) in Def. 3.2.1)**.** *Let Sec be as in Example 3.2.3. Define* $Q_1 = u_1.\mathbf{0} \boxplus u_2.\mathbf{0}$ *and* $Q_2 = o_1.\mathbf{0} \boxplus o_2.\mathbf{0}$. *Clearly,* Q_2 *provides* 0*-differential privacy. Consider now the parallel term* $Q_1 \,|\, Q_2$ *and define a scheduler that first executes a secret label* u *in* Q_1 *and then, if* u *is* u_1, *it selects* o_1, *while if* u *is* u_2, *it selects* o_2. *The rest proceeds like in Example 3.2.3.*

Example 3.2.5 (The need of condition (ii) in Def. 3.2.1)**.** *Let Sec be as in Example 3.2.3. Define* $Q_1 = o.\mathbf{0}$ *and* $Q_2 = u_1.(o_1.\mathbf{0} \oplus_{.5} o_2.\mathbf{0}) \boxplus u_2.(o_1.\mathbf{0} \oplus_{.5}$

$o_2.\mathbf{0}$). *It is easy to see that* Q_2 *provides* 0*-differential privacy. Consider the term* $Q_1 \mid Q_2$ *and define a scheduler that first executes a secret label* u *in* Q_2 *and then, if* u *is* u_1*, it selects first* Q_1 *and then the continuation of* Q_2*, while if* u *is* u_2*, it selects first the continuation of* Q_2 *and then* Q_1*. Hence, under this scheduler, for every sequence* $\vec{u} \in \mathcal{U}$*,* $p(oo_1|u_1\vec{u}) = p(oo_2|u_1\vec{u}) = 0.5$ *and also* $p(o_1o|u_2\vec{u}) = p(o_2o|u_2\vec{u}) = 0.5$ *while* $p(oo_1|u_2\vec{u}) = p(oo_2|u_2\vec{u}) = 0$ *and* $p(o_1o|u_1\vec{u}) = p(o_2o|u_1\vec{u}) = 0$*. Therefore* u_1 *and* u_2 *are disclosed.*

Intuitively, the existence of free observables (i.e. o *of* Q_1 *in this example) may create different interleavings, which can be used by the scheduler to mark different secrets.*

Example 3.2.6 (Replication operator does not preserve privacy.)**.** *Let Sec be as in Example 3.2.3. Define* $Q_1 = u_1.\mathbf{0} \boxplus u_2.\mathbf{0}$*,* $Q_2 = o_1.\mathbf{0} \boxplus o_2.\mathbf{0}$ *and* $Q = \tau.Q_1 \boxplus \tau.Q_2$*. Clearly,* Q *provides* 0*-differential privacy, because of the fact that both* Q_1 *and* Q_2 *are* 0*-differentially private and the compositionality property for non-deterministic choice in Theorem 3.2.2. Consider the replication* $!Q$ *and a scheduler that generates two copies* $Q \mid Q \mid !Q$*. For the first copy the scheduler selects* Q_1*, while for the second copy it selects* Q_2*. We have* $Q_1 \mid Q_2$ *which is already shown in Example 3.2.4 not preserve differential privacy.*

3.3 Trust and Legitimacy in Crowds

The compositionality property helps us to learn approximately a complex system from its sub-components or from its simplified evolutions. In this section, we extend the Crowds protocol with trust and legitimacy. More precisely, we consider

- *Member-wise trusted forwarders*: The member currently holding the message selects a forwarder only among the members which she thinks are trustable.
- *Legitimate initiators*: A member has the right to initiate requests only when she has been accepted as a "first-class citizen".

First we show, through some examples, that the trust network can influence dramatically the anonymity of a crowd. We argue that, due to this sensitivity to the local structure, it is difficult to reason globally about the network. Then we present that our compositional method, in contrast, does not rely on the symmetry condition, and provide an anonymity-preservation property. It expresses that the privacy of a crowd is bounded by the value of privacy of a simplified crowd, obtained by considering non-legitimate agents as attackers and therefore ignoring all the trust links starting from them. In particular, we formalize the trust and legitimacy extended Crowds in CCS_p code and we apply Theorem 3.2.2(4) to prove the anonymity-preservation property.

3.3.1 Examples

We show an example of a network in which changing a single trust link has a positive effect in the sense that it improves the level of anonymity, and another example in which such change has the opposite effect. We also explain how a channel matrix is affected when a member becomes non-legitimate. We show that the channel matrix in this case can be gotten for free, namely, obtained by just removing the row where this member is the initiator, without redoing any computations.

Consider an instance of a crowd where there are 4 members ($n = 4$). One of these members is an attacker ($c = 1$), and the others are honest ($m = 3$). For standard Crowds, it is easy to compute the level of differential privacy (cf. Section 3.1.4, Equation (3.1)). Assuming that $p_f = 4/5$ we have

$$p(o_i|u_i, \overline{NO}) = 1 - \frac{m-1}{n}p_f = 1 - \frac{2}{4}\cdot\frac{4}{5} = \frac{3}{5}$$

$$p(o_j|u_i, \overline{NO}) = \frac{1}{n}p_f = \frac{1}{4}\cdot\frac{4}{5} = \frac{1}{5} \qquad \text{for } i \neq j$$

The channel matrix is shown in Fig. 3.5, where the value $q = p(\overline{NO})$. The degree of differential privacy is $\epsilon_1 = \ln 3$.

We now consider the extension with trust information. We use arrows to indicate that the user at the starting point of an arrow trusts the one at

$p(o\|u)$	o_1	o_2	o_3	$\overline{OK}$
u_1	$\frac{3}{5}q$	$\frac{1}{5}q$	$\frac{1}{5}q$	$1-q$
u_2	$\frac{1}{5}q$	$\frac{3}{5}q$	$\frac{1}{5}q$	$1-q$
u_3	$\frac{1}{5}q$	$\frac{1}{5}q$	$\frac{3}{5}q$	$1-q$

Figure 3.5: The channel matrix for standard Crowds.

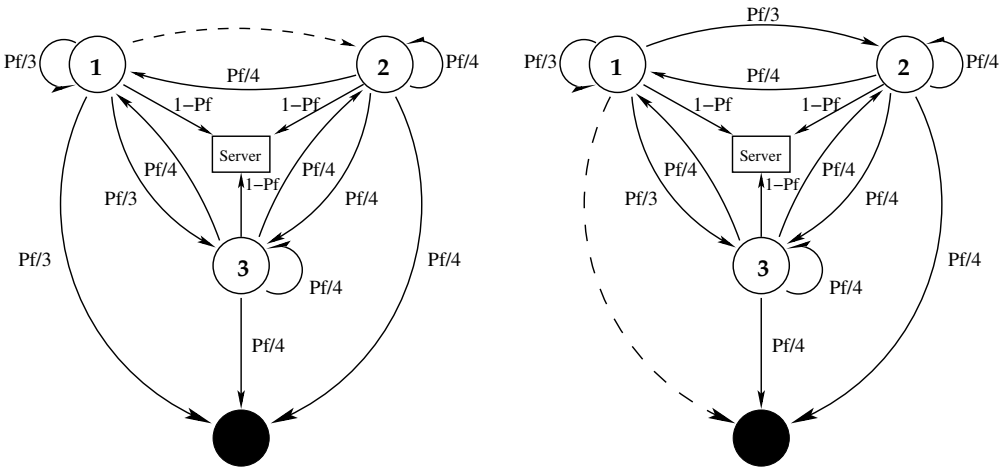

Figure 3.6: Two trust networks.

the end point. The square, the black circle and the white circles represent the server, the attacker and the honest members, respectively. Standard Crowds is a directed clique in the sense that from each node there are arrows pointing to all the other nodes, and also to itself. The two networks shown in Fig. 3.6 are obtained from standard Crowds by removing exactly one trust link. We denote by Crowd(a) and Crowd(b) the crowds in which the removed links are from user 1 to an honest member (i.e. $1 \to 2$) and from user 1 to an attacker (i.e. $1 \to \mathit{Attacker}$), respectively. Dashed lines are the removed links.

In order to compute the channel matrices, we use the *reachability analysis approach* [APvRS10], which consists in solving a system of linear equations derived from the graph. Take the columns $p(o_1|u_1)$, $p(o_1|u_2)$ and $p(o_1|u_3)$ in Crowd(a) as an example. We have the following equations, in which $p(o_1|f_i)$ represents the probability of observing member 1, given that

$p(o\|u)$	o_1	o_2	o_3	$\overline{OK}$
u_1	$\frac{15}{29}$	$\frac{1}{29}$	$\frac{4}{29}$	$\frac{9}{29}$
u_2	$\frac{5}{29}$	$\frac{10}{29}$	$\frac{15}{116}$	$\frac{41}{116}$
u_3	$\frac{5}{29}$	$\frac{11}{116}$	$\frac{11}{29}$	$\frac{41}{116}$

$p(o\|u)$	o_2	o_3	$\overline{OK}$
u_1	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{3}{5}$
u_2	$\frac{2}{5}$	$\frac{3}{20}$	$\frac{9}{20}$
u_3	$\frac{3}{20}$	$\frac{2}{5}$	$\frac{9}{20}$

Figure 3.7: The corresponding channel matrices.

there is a forwarder i on the path.

$$\begin{aligned} p(o_1|u_1) &= \tfrac{1}{3}p(o_1|f_1) + \tfrac{1}{3}p(o_1|f_3) + \tfrac{1}{3} \\ p(o_1|u_2) &= \tfrac{1}{4}p(o_1|f_1) + \tfrac{1}{4}p(o_1|f_2) + \tfrac{1}{4}p(o_1|f_3) \\ p(o_1|u_3) &= \tfrac{1}{4}p(o_1|f_1) + \tfrac{1}{4}p(o_1|f_2) + \tfrac{1}{4}p(o_1|f_3) \end{aligned}$$

$$\begin{aligned} p(o_1|f_1) &= p_f\, p(o_1|u_1) \\ p(o_1|f_2) &= p_f\, p(o_1|u_2) \\ p(o_1|f_3) &= p_f\, p(o_1|u_3) \end{aligned}$$

By solving the system we obtain the channel matrix shown in Fig. 3.7(a), which is differentially private with $\epsilon_2 = \ln 10$. The channel matrix for Crowd(b), shown in Fig. 3.7(b), can be computed analogously. The corresponding privacy level is $\epsilon_3 = \ln \frac{8}{3}$.

Fig. 3.7(a) shows that after the link $1 \to 2$ is discarded, the anonymity of user 2 is compromised dramatically. Fig. 3.7(b) shows that, on the other hand, if we discard a link to an attacker ($1 \to \mathit{Attacker}$), the anonymity of the whole crowd gets improved. We see that changing one single trust link can have a huge impact on the level of privacy, in both directions.

The following example says that the channel matrix for non-legitimate members can be obtained by removing the rows in which non-legitimate members are initiators, rather than re-computing the system of linear equations.

Example 3.3.1. *Consider the two trust networks in Fig. 3.6. Assume that user* 1 *is a non-legitimate member. She is not allowed to initiate the*

$p(o\|u)$	o_1	o_2	o_3	$\overline{OK}$
u_2	$\frac{5}{29}$	$\frac{10}{29}$	$\frac{15}{116}$	$\frac{41}{116}$
u_3	$\frac{5}{29}$	$\frac{11}{116}$	$\frac{11}{29}$	$\frac{41}{116}$

$p(o\|u)$	o_2	o_3	$\overline{OK}$
u_2	$\frac{2}{5}$	$\frac{3}{20}$	$\frac{9}{20}$
u_3	$\frac{3}{20}$	$\frac{2}{5}$	$\frac{9}{20}$

Figure 3.8: When user 1 is non-legitimate in Crowd(a) and (b).

message. When we solve the system of linear equations derived from the graph, the equations for computing the probabilities $p(o_i|u_1)$ $(i \in \{1,2,3\})$ *are removed, and this removal does not affect the values of the remaining probabilities. (For example, the removal of the equation of* $p(o_1|u_1)$ *does not change the values of* $p(o_1|u_2)$, $p(o_1|u_3)$ *and* $p(o_1|f_i)$ $(i \in \{1,2,3\})$*). Hence the channel matrix when user* 1 *is a non-legitimate member is obtained by simply removing the row of* u_1 *from the previous channel matrix. After the removal of the rows of* u_1*, the channel matrices in Fig. 3.7 become the two matrices shown in Fig. 3.8 of which the privacy levels* $\epsilon'_2 = \ln \frac{40}{11}$ *and* $\epsilon'_3 = \ln \frac{8}{3}$*, respectively.*

Vice versa, when user 1 *enjoys legitimacy again, we just add back the row of* u_1 *to the channel matrix, in which* $p(o_i|u_1) = p(o_i|f_1)/p_f$*.*

3.3.2 The CCS$_p$ code for the extended Crowds protocol

The extended Crowds protocol with member-wise trusted forwarders and legitimacy information expressed in CCS$_p$ is stated in Fig. 3.9. For simplicity, we introduce a notation for value-passing in CCS$_p$, following standard lines.

$$\begin{array}{rrcl} \textit{Input} & a(i).Q & = & \lfloor + \rfloor_j \, a_j.Q[j/i] \\ \textit{Output} & \bar{a}\langle i \rangle & = & \bar{a}_i \end{array}$$

We use $\bigoplus^{\mathcal{U}}$ to represent a uniform distribution. Label u_i represents that the initiator is user i. Label $\bar{a}_j\langle i \rangle$ describes that member i forwards the message to member j. Label $\overline{OK}$ means that the request of initiator is successfully sent to the server, while label o^i_j represents that the attacker i detects a

$$
\begin{array}{rcl}
\mathit{Initiator} & = & \bigoplus^{\mathcal{U}}_{i\in H_l} p_i.u_i.(\bigoplus^{\mathcal{U}}_{j\in T_i} p_j.\bar{a}_j\langle i\rangle) \\
\mathit{Honest}_i & = & a_i.((\bigoplus^{\mathcal{U}}_{j\in T_i} p_j.\bar{a}_j\langle i\rangle.\mathit{Honest}_i) \oplus_{p_f} \overline{d}\,) \\
\mathit{Attacker}_i & = & a_i(j).o^i_j \\
\mathit{Server} & = & d.\overline{OK} \\
\mathit{Members}(H,B) & = & \mathit{Server}\,|\,\mathit{Initiator}|\prod_{i\in H}\mathit{Honest}_i|\prod_{j\in B}\mathit{Attacker}_j \\
\mathit{Crowds}(H,B) & = & (\nu d)(\nu a_1,a_2,\ldots,a_n)\mathit{Members}(H,B)
\end{array}
$$

Figure 3.9: A variant of Crowds with trust and legitimacy information.

message from the honest member j. We denote by T_i the subset of crowd members which the i-th honest member trusts, H_l the set of legitimate honest members. Clearly, $H_l \subseteq H$. The set of secret labels is $\{\, u_i \,|\, i \in H_l\}$ and the set of observable labels is $\{\, \overline{OK}\,\} \cup \{\, o^i_j \,|\, i \in B, j \in H\,\}$.

Term *Initiator* first generates an initiator as per the uniform distribution over all members in the crowd, and then sends the request to a forwarder chosen as well according to the uniform distribution over all members in the crowd. We refer to Section 3.1.4 for the meaning of the term Honest_i and the term $\mathit{Attacker}_i$. The two parameters H and B in $\mathit{Members}(H,B)$ and $\mathit{Crowds}(H,B)$ specify respectively the set of honest members and of attackers in the crowd. Our extension lies in that an initiator is probabilistically chosen from H_l rather than H, and a next forwarder is probabilistically chosen from T_i rather than C.

3.3.3 An anonymity-preservation property

Consider a crowd with $n+1$ members. Assume that the agent $n+1$ just joined the crowd but she does not enjoy the legitimacy to be an initiator right away, namely, she is non-legitimate. Applying our compositionality theory, we show that the privacy of this crowd is bounded by the value of privacy of a simplified crowd, which is obtained by considering the non-legitimate agent $n+1$ as an attacker and therefore ignoring all the trust

$$\begin{aligned} \mathit{Members}(H \cup \{n+1\}, B) &= \mathit{Members}(H, B) \,|\, \mathit{Honest}_{n+1} \\ \mathit{Crowds}(H \cup \{n+1\}, B) &= (\nu d)(\nu a_1, a_2, \ldots, a_{n+1})\mathit{Members}(H \cup \{n+1\}, B) \end{aligned}$$

Figure 3.10: Specification of the addition of a honest agent $n+1$.

links starting from her. The fact is supported by Theorem 3.3.2. Note that its proof does not rely on the symmetry conditions.

The process term representing the addition of honest agent $n+1$ to the crowd of n participants, $\mathit{Crowds}(H \cup \{n+1\}, B)$, is presented in Fig.3.10. Basically, it is a parallel composition of the process term $\mathit{Members}(H, B)$ of the old crowd and the process term Honest_{n+1} of agent $n+1$. We denote the simplified crowd by the process term $\mathit{Crowds}(H, B \cup \{n+1\})$, which is constructed in a similar way to $\mathit{Crowds}(H \cup \{n+1\}, B)$: instead of being paralleled with Honest_{n+1}, the term $\mathit{Members}(H, B)$ is paralleled with $\mathit{Attacker}_{n+1}$. We omit its code for simplicity.

Theorem 3.3.2. *$dp[\![\mathit{Crowds}(H \cup \{n+1\}, B)]\!] \leq dp[\![\mathit{Crowds}(H, B \cup \{n+1\})]\!]$, where the user $n+1$ is non-legitimate.*

Proof. Consider the term $\mathit{Crowds}(H \cup \{n+1\}, B)$. Remove the term Honest_{n+1} and the corresponding restrictions on the labels of Honest_{n+1}. Let Q be the process term obtained in this way. Note that the free labels through which the old crowd communicates with agent $n+1$ are $\{d\} \cup \{a_j | j \in T_{n+1}\} \cup \{\bar{a}_{n+1}\langle i \rangle | i \in S_{n+1}\}$, where S_{n+1} is the subset of crowd members who trust the agent $n+1$. The labels $\bar{a}_{n+1}\langle i \rangle$ are now observable and give the same information as if the agent $n+1$ were an attacker. Because they reveal the identity of member i who is sending the message. Furthermore, the labels a_j give the same information as if the agent $n+1$ were an attacker continuing to forward the message. Because they activate the process term of user j and the protocol proceeds. The channel matrix we obtain from Q is isomorphic to the channel matrix of $\mathit{FCrowds}(H, B \cup \{n+1\})$: the only difference is that each column o_i^{n+1} is now renamed to $\bar{a}_{n+1}\langle i \rangle$. Thus

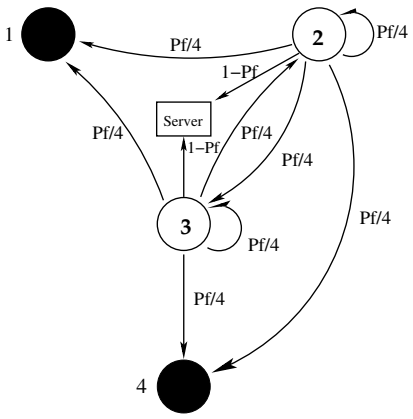

Figure 3.11: The simplified trust network when user 1 becomes an attacker.

$p(o\|u)$	o_2^1	o_3^1	o_2^4	o_3^4	$\overline{OK}$
u_2	$\frac{1}{3}$	$\frac{1}{12}$	$\frac{1}{3}$	$\frac{1}{12}$	$\frac{1}{6}$
u_3	$\frac{1}{12}$	$\frac{1}{3}$	$\frac{1}{12}$	$\frac{1}{3}$	$\frac{1}{6}$

Figure 3.12: The corresponding channel matrix.

$dp[\![Q]\!] = dp[\![FCrowds(H, B \cup \{n+1\})]\!]$. By Proposition 3.1.11, we have $dp[\![Q]\!] = dp[\![Crowds(H, B \cup \{n+1\})]\!]$.

Now we add back the term $Honest_{n+1}$. Since, user $n+1$ will not be chosen as an initiator (because she is not legitimate), the secret label a_{n+1} will not be used. Furthermore $Honest_{n+1}$ does not contain any other secret labels. Therefore,

$$(\nu d)(\nu a_1, a_2, \ldots, a_{n+1}) Honest_{n+1}$$

is a safe component. By Theorem 3.2.2(4), we have:

$$dp[\![Crowds(H \cup \{n+1\}, B)]\!] \leq dp[\![Q]\!] = dp[\![Crowds(H, B \cup \{n+1\})]\!]$$

which concludes the proof. □

The following example shows that the above theorem indeed gives a privacy bound for the two crowds as shown in Fig. 3.6.

Example 3.3.3 (Crowd(a) and (b) in Fig. 3.6 revisited)**.** *Recall the two trust networks in Fig. 3.6. Assume that user* 1 *is a non-legitimate member. In this case, the privacy levels of Crowd(a) and (b) have already been shown in Fig. 3.8, with* $\epsilon'_2 = \ln \frac{40}{11}$ *and* $\epsilon'_3 = \ln \frac{8}{3}$*, respectively. By considering user* 1 *as an attacker and removing all the trust links starting from her, we get a simplified trust network shown in Fig. 3.11 and its corresponding channel matrix with privacy level* $\epsilon_s = \ln 4$ *in Fig. 3.12. (The simplified crowd obtained from Crowd(a) is the same as the simplified one from Crowd(b).)*

By applying Theorem 3.3.2, we know that given that user 1 *is a non-legitimate member,* ϵ_s *is an upper bound for the privacy levels of Crowd(a) and (b), as it is also exemplified by the fact that* $\epsilon'_2 \leq \epsilon_s$ *and* $\epsilon'_3 \leq \epsilon_s$*.*

3.4 Degradation of privacy by trust

In this section, we first show through a small example that in the presence of trust information, the anonymity of crowd members may get harmed badly. In these cases the leak of information is brought in by the trust information rather than the fault of the Crowds protocol. Thus we introduce a notion of *trustworthy adjacency* relation to rule out this factor, thus retrieving a measure of real privacy. Furthermore, we show how this notion can help to indicate the cases in which the overestimation in Theorem 3.3.2 is a false negative, namely, the privacy of a crowd may be very well protected while the estimation says that the crowd breaks the privacy entirely.

3.4.1 An adjacency relation based on trust

The following example shows that privacy is broken in Crowds due to the trust information.

Example 3.4.1. *Consider the crowd shown in Figure 3.13. It consists of four honest members* $1, 2, 3, 4$ *and two attackers* 5 *and* 6*. Since no trust link points from users* 1 *and* 2 *to users* 3 *and* 4*, when user* 1 *and user* 2 *are the initiators, the probabilities of detecting user* 3 *and user* 4 *are zero. Thus we will have the column of the detection* o_3^6*, in which* $p(o_3^6|u_1) = p(o_3^6|u_2) = 0$*,*

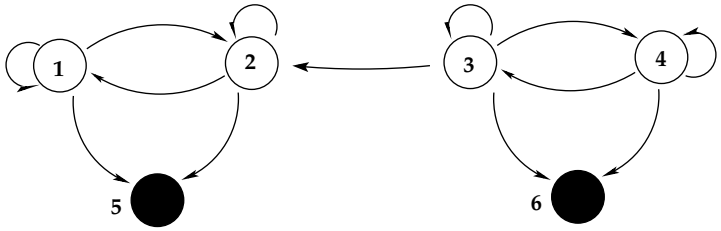

Figure 3.13: A crowd of which privacy is broken due to trust.

but $p(o_3^6|u_3) \neq 0$ *and* $p(o_3^6|u_4) \neq 0$*, which breaks the privacy. Analogously, this problem exists also in the column of* o_4^6*.*

Note that, users 1 *and* 2 *are reachable from each other, but they both can not reach users* 3 *and* 4*. Rather than assuming that the identity of each user is adjacent to any other, we would like to take into account the trust information, and consider users* 1 *and* 2 *as a pair of close secret identities, i.e., adjacent, but users* 1 *and* 3 *as a pair of far identities, i.e., not adjacent, and using this fine notion of adjacency relation to measure the value of differential privacy.*

Now we formalize the notion of adjacency relation illustrated in the above examples. Intuitively, we consider as adjacent two users who can reach a same set of members that are directly exposed to attackers. We need to introduce some notations first. We denote by $R(i)$ the set of identities of crowd members that can be reached by user i. In Fig. 3.13, we have $R(1) = R(2) = \{1, 2, 5\}$ and $R(3) = R(4) = \{1, 2, 3, 4, 5, 6\}$. We denote by S_i the subset of identities of crowd members who trust user i, i.e. the members who can reach user i in one step. We have $S_5 = \{1, 2\}$, user 3 cannot reach directly the attacker 5, and $S_6 = \{3, 4\}$. Let $W = \cup_{k \in B} S_k$, it represents the set of honest members that are exposed to attackers, thus forms the observable detections. We have $W = S_5 \cup S_6 = \{1, 2, 3, 4\}$.

Definition 3.4.2 (Trustworthy adjacency)**.** *Given a crowd with trust information, two honest members* i, j *are* trustworthily adjacent *to each other, iff* $R(i) \cap W = R(j) \cap W$*.*

Proposition 3.4.3. *If in a crowd users i and j are not trustworthily adjacent, then under the adjacency relation defined in terms of a clique, the crowd does not preserve differential privacy.*

Proof. Since $R(i) \cap W \neq R(j) \cap W$, there must exist at least a honest member l and an attacker k, such that $l \in S_k$, $l \in R(i)$ but $l \notin R(j)$. When user i initiates the request, the message has a probability of being forwarded to member l, probably resulting in member l being detected by attacker k. It is easy to see that in the channel matrix, we have $p(o_l^k|u_i) \neq 0$ but $p(o_l^k|u_j) = 0$. Straightforwardly, the crowd does not provide differential privacy. □

From the above proposition, we see clearly the case in which the privacy is broken due to the comparison of conditional probabilities between two users that are not trustworthily adjacent. Therefore, for Crowds with trust information, we propose using the trustworthy adjacency relation, rather than the generic adjacency relation for anonymity as previously defined in terms of a clique (cf. Example 2.6.2), to measure the value of differential privacy, thus eliminating the factor of the connectivity induced by the trust links.

3.4.2 False negatives in Theorem 3.3.2

Theorem 3.3.2 offers an upper bound for a crowd containing trust and legitimacy information, by considering the case in which non-legitimate members are treated as attackers. It may give an infinity upper bound. As shown in the next example, the estimation in this case is a false negative, namely, the privacy of a crowd is actually a bounded value rather than infinity.

Example 3.4.4. *Consider the crowd (a) in Figure 3.14, in which there are two honest members* $1, 2$*, one non-legitimate member* 3 *and one attacker. Observing the trust information, we know that when either user* 1 *or user* 2 *is detected by the attacker, both member* 1 *and member* 2 *could be the initiator. There does not exist the case in which both zero and non-zero*

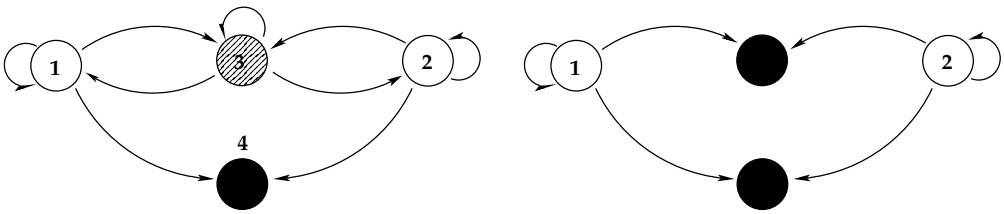

Figure 3.14: A crowd before and after the simplification.

probabilities occur in the same column of the channel matrix. Thus we will get a finite value of differential privacy for this crowd.

While applying the transformation method of Theorem 3.3.2, considering user 3 *as an attacker and ignoring the links starting from her, we get a simplified crowd as shown in (b). Now users* 1 *and* 2 *are isolated from each other. When an attacker gets the message from user* 1 *(resp.* 2*), she will know for sure that user* 1 *(resp.* 2*) is the initiator. Thus the privacy is completely revealed, and the resulting estimation announces a false negative.*

Now we show how to find the cases in which the estimation given in Theorem 3.3.2 is a false negative. The following proposition says that if the transformation of Theorem 3.3.2 makes two honest users, which used to be trustworthily adjacent, afterward not trustworthily adjacent, then the estimation will be an infinity value.

Proposition 3.4.5. *For Theorem 3.3.2, if two honest users* i *and* j*,* $(i, j \neq n+1)$ *are trustworthily adjacent in Crowds*$(H \cup \{n+1\}, B)$*, while not trustworthily adjacent in Crowds*$(H, B \cup \{n+1\})$*, then Crowds*$(H, B \cup \{n+1\})$ *does not satisfy differential privacy.*

It is easily obtained from a similar analysis to Proposition 3.4.3.

Let us revisit Example 3.4.4. Before applying the transformation, users 1 and 2 are reachable from each other through the connection links offerred by member 3, however this is not the case any more after applying the transformation. Because the transformation changes the adjacency relation between users 1 and 2. By the above proposition, we know that when

applying Theorem 3.3.2 in this case, we will get the answer that the privacy is not preserved in this crowd.

3.5 Users' preference levels in Crowds

In [HPSE10], the authors notice that in the context of Crowds, the message forwarded in the network reveals also the identity of the end sever. This additional observation may leak additional information about the initiator of the transaction when users' habits of Web browsing is not uniform, which is usually the case in the real world. In this section we analyze the impact of these additional observables on the security of a protocol in the context of differential privacy.

In [HPSE10] the authors express this additional information about the end server in terms of a random variable $\mathcal{E}$, whose values $e_1, \ldots, e_l$ are assumed to be observable, and the conditional probabilities $p(e_k|u_i)$, that is, the correlation between these additional observables and the legitimate initiators, is publicly known. Here the conditional probability $p(e_k|u_i)$ expresses the probability that the user i visits the server k when she initiates a transaction. In other words, the probabilities $p(e_k|u_i)$ define a channel matrix modeling the users' profiles of Web browsing.

In [HPSE10], it is shown that in the context of normal Crowds, the observables $\mathcal{O}$ and the additional observables $\mathcal{E}$ are independent for every initiator. Here again we shall assume this conditional independency, though the choice of a forwarder may depend on the trust links (e.g. exit policies of Tor network relays)[2]. We show that in the context of differential privacy, the knowledge of users' habits reduces the privacy of the protocol by a factor that is less than or equals to the privacy level of the channel matrix $p(e_k|u_i)$ modeling the users' profiles of Web browsing. Formally, let $dp[\![C_{\mathcal{O}}]\!]$, $dp[\![C_{\mathcal{E}}]\!]$, and $dp[\![C_{\mathcal{O},\mathcal{E}}]\!]$ denote respectively the privacy levels of the protocol as expressed in the previous sections, and the channel $p(e_k|u_i)$ model the users' profiles of Web browsing. The protocol where both observables are

[2]We leave this general case for future work.

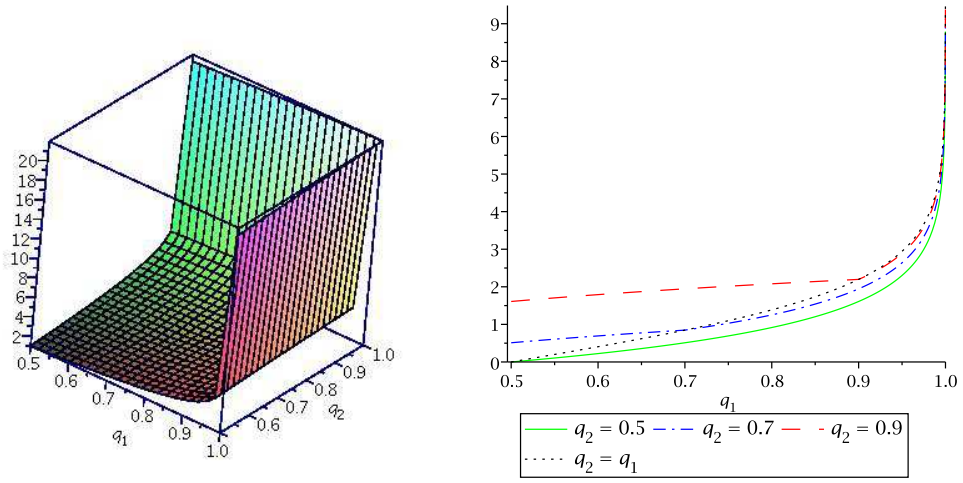

Figure 3.15: Impact of the knowledge of users' profiles on privacy in Crowds.

considered is modeled by the channel $p(o_j, e_k|u_i)$. Then the following holds. This is actually a particular case of a well known property of differential privacy [McS09], but the formulation is a bit different, so we detail the proof here.

Theorem 3.5.1. $dp[\![C_{\mathcal{O},\mathcal{E}}]\!] \leq dp[\![C_{\mathcal{O}}]\!] + dp[\![C_{\mathcal{E}}]\!]$.

Proof. Let $\epsilon_{ex} = dp[\![C_{\mathcal{O},\mathcal{E}}]\!]$, $\epsilon = dp[\![C_{\mathcal{O}}]\!]$ and $\epsilon' = dp[\![C_{\mathcal{E}}]\!]$. Thus we have $e^{\epsilon_{ex}} = \max_{j,k,i,i'} \dfrac{p(o_j, e_k|u_i)}{p(o_j, e_k|u'_i)}$, $e^{\epsilon} = \max_{j,i,i'} \dfrac{p(o_j|u_i)}{p(o_j|u'_i)}$, and $e^{\epsilon'} = \max_{j,k,k'} \dfrac{p(e_k|u_i)}{p(e_k|u'_i)}$. But for every j, k, i, and i', since $\mathcal{O}$ and $\mathcal{E}$ are conditionally independent,

$$\frac{p(o_j, e_k|u_i)}{p(o_j, e_k|u'_i)} = \frac{p(o_j|u_i) \cdot p(e_k|u_i)}{p(o_j|u'_i) \cdot p(e_k|u'_i)} = \frac{p(o_j|u_i)}{p(o_j|u'_i)} \cdot \frac{p(e_k|u_i)}{p(e_k|u'_i)}$$

Hence by taking the max, we have $e^{\epsilon_{ex}} \leq e^{\epsilon} \cdot e^{\epsilon'}$ which concludes the proof. □

Example 3.5.2 (Crowds with users' preference for servers)**.** *Here we illustrate the impact of the users' preference levels on the security of the crowd in the setting of differential privacy (cf. Theorem 3.5.1).*

Consider an instance of a crowd where there are two destination servers e_1, e_2*, and three honest members* u_1, u_2, u_3*. Assume also that users' preferences for servers are as follows:*

$$p(e|u) = \begin{cases} q_1 & u \in \{u_1, u_2\}, e = e_1 \\ 1 - q_1 & u \in \{u_1, u_2\}, e = e_2 \\ 1 - q_2 & u = u_3, e = e_1 \\ q_2 & u = u_3, e = e_2 \end{cases}$$

where $0.5 \leq q_1, q_2 \leq 1$*. In other words, users 1 and 2 communicate more often with server 1. Similarly user 3 prefers server 2. In this setting, it is easy to see that the impact of the users' profiles is* $dp[\![C_{\mathcal{E}}]\!] = \ln\left(\max\{\frac{q_1}{1-q_2}, \frac{q_2}{1-q_1}\}\right)$.

Figure 3.15 shows the impact factor $dp[\![C_{\mathcal{E}}]\!]$ *when the users' preferences for servers vary from* 0.5 *to* 1*. It clearly shows that when both* q_1 *and* q_2 *are equal to* 0.5*, that is, when both servers are equally preferred by every user, the impact is null. However, as soon as some users' preference starts biasing towards one server, their privacy also starts deteriorating. A preference strongly biased towards one server will definitely break the privacy, as it is strong enough to distinguish the users preferring it from the others.*

3.6 Related work

- *Compositionality properties of probabilistic process calculi for security protocols.* In [DPW06] Deng et al. use the notion of relative entropy to measure privacy. In [BCP08, CPB] Braun et al. consider the safety measured by Bayes risk. The compositionality results in this chapter are closely related to those of [BCP08, CPB], although we use a different measure of protection (differential privacy).

- *Compositionality of Differential Privacy.* As already stated, there is a vast body of work on formal methods for differential privacy. Composi-

tional methods, as one of the most important features, have been intensively investigated in the field of statistical databases [McS09] and programs [BKOB12, RP10]. These works investigate the so-called sequential and parallel compositions of queries (programs), which, in their context, mean a sequence of queries (programs) applied to the same dataset and to disjoint parts of the dataset, respectively. Under this setting, they have proved that the sequential composition decreases the privacy, and the parallel composition maintains the privacy. Our result about the replication and the parallel composition in CCS_p are reminiscent of the above results. But the context is different. In particular, the parallel composition concerned in this chapter is different from the above one, in that the parallel operator here represents interactions of concurrent agents. Our restrictions on the parallel composition are incomparable with those of [McS09, BKOB12, RP10] (disjointness of databases).

- *Other extensions on Crowds.* In [HPSE10] the authors study the impact of additional information attackers gather before attacking the protocol. In [SEH10] Sassone et al. extend Crowds by considering a trust distribution over participants which is not uniform but the same from all members' points of view, and therefore can still be thought of as the symmetry condition. In [SHY10] they consider a more realistic and complicated scenario in which the forwarding policies are neither uniform nor the same for every member, however no general privacy-preserving property is given under this scenario. By checking the proof of Theroem 3.3.2 in this chapter, it holds regardless of whether the forwarding distributions are uniform or not.

3.7 Conclusion

In this chapter, we have defined the degree of differential privacy for concurrent systems expressed in a probabilistic process calculus, and investigated how the privacy is affected under composition of the CCS_p constructs. We have shown a proof for the fact that in Crowds, stopping or continuing

forwarding the message after the first detection is the same from the point of view of differential privacy. After pointing out the close relation between privacy and anonymity, we have applied our compositionality result to prove an anonymity-preservation property of an extended Crowds protocol with legitimate initiators and member-wise trusted forwarders. We have considered also the case in which some leak of privacy is induced by the trust information, and we have introduced the notion of trustworthy adjacency relation to exclude this factor. Finally, we have shown that the impact of users' preference levels studied in [HPSE10] can be expressed as a degradation factor that depends only on the privacy level of the channel modeling the source of the additional observation.

One immediate task is how to compute or verify the levels of privacy of processes, thus combining our compositional method to obtain the level of privacy of a compound program. In the next chapter of the thesis, we will address this issue by developing a quantitative bisimulation-based verification approach that can track global privacy leakage from local information.

Four

Bisimulations for Differential Privacy

In this chapter we deal with the problem of verifying differential privacy properties for concurrent systems, modeled as probabilistic automata admitting both nondeterministic and probabilistic behavior. In such systems, reasoning about the probabilities requires *solving* the nondeterminism first, and to such purpose the usual technique is to consider functions, called *schedulers*, which select the next step based on the history of the computation. However, in our context, as well as in security in general, we need to restrict the power of the schedulers and make them unable to distinguish between secrets in the histories, or otherwise they would plainly reveal them by their choice of the step. See for instance [CCK+06, CNP09, CP10, APSVR11] for a discussion on this issue. Thus we consider a restricted class of schedulers, called *admissible schedulers*, following the definition of [APSVR11]. Admissibility is introduced to deal with bisimulation-like notions in security contexts: Two bisimilar processes are typically considered to be indistinguishable, yet an unrestricted scheduler could trivially separate them.

The property of differential privacy requires that the observations generated by two different secret values be probabilistically similar. In standard concurrent systems the notion of similarity is usually formalized as an equivalence, preferably preserved under composition, i.e., a congruence. We mention in particular trace equivalence and bisimulation. The first is often

used for its simplicity, but in general is not compositional [JS90]. The second one is a congruence and it is appealing for its proof technique. Process equivalences have been extensively used to formalize security properties like secrecy [AG99] and noninterference [FG00, RS01, Smi03].

In this chapter we focus on approximate bisimulations suitable for verifying differential privacy. Namely, bisimulations for which the degree of similarity between two processes determines an upper bound on the ratio of the probabilities of the respective observables. We start by considering the framework proposed by Tschantz *et al.* [TKD11], which was explicitly designed for the purpose of verifying differential privacy. Their verification technique is based on proving the existence of an indexed family of bijections between states. The parameter of the starting states, representing the privacy budget, determines the level of differential privacy of the system, which decreases over time by subtracting the absolute difference of probabilities in each step during mutual simulation. Once the balance reaches zero, processes must behave exactly the same. We reformulate this notion in the context of probabilistic automata, showing its metrical properties.

The above technique is sound, but has a rather rigid budget management. The goal of this chapter is to make the technique more permissive by identifying an approximate bisimulation that is more relaxed and still implies an upper bound on the privacy leakage.

In particular, the bisimulation we propose is based on a thriftier use of the privacy budget, which is inspired by the notion of *amortisation* used in [KAK05, dFERVGR07]. We name it *ϵ-amortised probabilistic bisimulation*, where ϵ depicts the degree of similarity between processes. The idea is that, when updating the variation of privacy leakage, the differences among the probabilities of related states are kept with their sign, and added with their sign through each step. In this way, successive differences can compensate (amortise) each other, and rather than always being consumed, the privacy budget may also be refurbished. In [KAK05] the idea of amortisation is applied on a set of cost-based actions. The quantitative feature considered here is discrete probability distributions over states, which is shown to benefit from the theory of amortisation as well.

Furthermore, the bijection condition in [TKD11] is admitted to be too strict, as, for instance, the resulting relation is not substitutive under process combinators; its 0-indexed states fail to fully characterize the standard probabilistic bisimilarity; and the weak transitions built on it are not transitive. In our work, the bijection requirement is removed, to allow the split of probabilities when relations between processes are lifted to distributions over processes, so that these undesirable consequences are avoided.

In particular, we prove that our amortised bisimulation is substitutive under several process combinators including parallel composition. We show that the 0-indexed states in the our notion fully characterise bisimilarity. For the weak transitions built on it, we will show in the next chapter that they are transitive.

Finally, we illustrate the verification technique of differential privacy using the example of the Dining Cryptographers Problem (DCP) with biased coins.

Contribution. The main contributions of this chapter can be summarized as follows:

- We reformulate the notion of approximate similarity proposed in [TKD11] in the context of probabilistic automata and we study its metrical properties (in Section 4.2).

- We propose the ϵ-amortised bisimulation which is more liberal than the former one, in such a way that the total differences of probabilities get amortised during the mutual simulation, and the split of probabilities is allowed in the lifting operation. We show that the level of differential privacy is continuous with respect to the ϵ-amortised bisimulation, which says that if every two processes running on two adjacent secrets of a system are close in the bisimulation then the system is differentially private, making our notion suitable for verification (in Section 4.3).

- We show that 0-amortised bisimulation fully characterizes bisimilarity (in Section 4.4).

- We present that our amortised bisimulation is substitutive under typical CCS_p operators (in Section 4.5).

- We use the verification framework to show that the Dining Cryptographers protocol with probability-p biased coins is $|\ln \frac{p}{1-p}|$-differentially private. (in Section 4.6).

In the next section we introduce some preliminary notions that shall be used in this chapter, in particular, the definition of differential privacy under admissible schedulers.

4.1 Preliminaries

4.1.1 Admissible scheduler

We consider a restricted class of schedulers, called *admissible schedulers*, following the definition of [APSVR11]. Essentially this definition requires that whenever given two *adjacent* states s, s', namely, differing only for the choice for some secret value, then the choice made by the scheduler on s and s' should be consistent, i.e. the scheduler should not be able to make a different choice on the basis of the secret. Note that in [TKD11] scheduling is not an issue since non-determinism is not allowed.

More precisely, in [APSVR11] admissibility is achieved by introducing tags for transitions. Admissible schedulers are viewed as entities that have access to a system through a screen with buttons, where each button represents one (current) available option, i.e. an enabled tag. A scheduler ζ is admissible if for all finite executions having the same sequence of screens, ζ decides the same tagged transition for them.

More formally, given a PA $\mathcal{A}$, we define an operation $t : Exec^*(\mathcal{A}) \to A^*$ on executions that maps an execution to its labels.

Definition 4.1.1. *Let $\alpha = s_0 a_0 s_1 a_1 \cdots a_n s_{n+1}$ be a finite execution of a PA $\mathcal{A}$, then we define t as:*

$$t(\alpha) = a_0 a_1 \cdots a_n.$$

Define another operation $l : D \to A$, where D is the transition relation, as for $s \xrightarrow{a} \mu \in D$, $l(s \xrightarrow{a} \mu) = a$, then we are ready to give the definition of admissible schedulers as follows.

Definition 4.1.2 (Admissible Schedulers)**.** *A scheduler ζ is admissible if for all executions $\alpha, \alpha' \in Exec(\mathcal{A})$*

$$t(\alpha) = t(\alpha') \text{ implies } l(\zeta(\alpha)) = l(\zeta(\alpha')).$$

In notation, $\mathcal{A}_\zeta$ the fully probabilistic automaton (defined in Section 2.2) obtained from $\mathcal{A}$ where all the non-determinism is resolved by ζ. Given a trace $\vec{t}$ and an execution α, we add a script ζ and use $\Pr_\zeta[\alpha \rhd C_{\vec{t}}]$ to denote the probability of a cone $C_{\vec{t}}$ induced by α, given the scheduler ζ. Eq. 2.1 can be easily rephrased as follows:

$$\Pr_\zeta[\alpha \rhd C_{\vec{t}}] = \begin{cases} 1 & \text{if } \vec{t} = [\,], \\ 0 & \text{if } \vec{t} = a^\frown \vec{t'} \text{ and } act(\zeta(\alpha)) \neq a, \\ \sum_{s_i} \mu(s_i) \Pr_\zeta[\alpha a s_i \rhd C_{\vec{t'}}] & \\ \qquad \text{if } \vec{t} = a^\frown \vec{t'} \text{ and } \zeta(\alpha) = s \xrightarrow{a} \mu. & \end{cases} \tag{4.1}$$

4.1.2 Differential privacy under admissible scheduler

We have introduced the notion of differential privacy [Dwo06] in Section 2.6. It was originally defined in the context of statistical databases, by requiring that a mechanism (i.e. a probabilistic query) gives similar answers on *adjacent* databases, that is those differing on a single row.

In this chapter, we study concurrent systems taking a secret as input and producing an observable trace as output. Let U be a set of secrets and $\smile$ an adjacency relation on U, where $u \smile u'$ denotes the fact that two close secrets u, u' should not be easily distinguished by the adversary after seeing observable traces. A *concurrent system* $\mathcal{A}$ is a mapping of secrets to probabilistic automata, where $\mathcal{A}(u), u \in U$ is the automaton modelling the behavior of the system when running on u. Differential privacy can be directly adapted to this context:

Definition 4.1.3 (Differential Privacy under admissible scheduler)**.** *A concurrent system $\mathcal{A}$ satisfies* ϵ-differential privacy *(DP) iff for any $u \backsim u'$, any finite trace $\vec{t}$ and any admissible scheduler ζ:*

$$\Pr_\zeta [\mathcal{A}(u) \rhd C_{\vec{t}}\,] \leq e^\epsilon \cdot \Pr_\zeta [\mathcal{A}(u') \rhd C_{\vec{t}}\,]$$

4.2 The accumulative bisimulation

In this section, we present a probabilistic bisimulation, *accumulative bisimulation*, based on a reformulation of the relation family proposed by Tschantz *et al.* in [TKD11] and exhibit its metrical properties.

We start by defining an approximate lifting operation that lifts a relation over states to a relation over distributions. Intuitively, we use a parameter ϵ to represent the total privacy leakage budget. A parameter c ranging over $[0, \epsilon]$, starting from 0, records the current amount of leakage and increasing over time by adding the maximum absolute difference of probabilities, denoted by σ, in each step during mutual simulation. Once c reaches the budget bound ϵ, processes must behave exactly the same. Since the total bound is ϵ, only a total of ϵ privacy can be leaked, a fact that will be used later to verify differential privacy. We use T (short for Tschantz *et al.*) to simply differentiate notions of this section from the following sections.

Definition 4.2.1. *Let $\epsilon \geqslant 0$, $c \in [0, \epsilon]$, $\mathcal{R} \subseteq S \times S \times [0, \epsilon]$. The* T-lifting *of $\mathcal{R}$ up to c, denoted by $\mathcal{L}^T(\mathcal{R}, c)$, is the relation on $Disc(S)$ defined as:*

$$\mu \mathcal{L}^T(\mathcal{R}, c) \mu' \quad \textit{iff} \quad \exists \textit{ bijection } \beta : supp(\mu) \to supp(\mu') \textit{ such that}$$
$$\forall s \in supp(\mu) : (s, \beta(s), c + \sigma) \in \mathcal{R} \quad \textit{where} \quad \sigma = \max_{s \in supp(\mu)} |\ln \frac{\mu(s)}{\mu'(\beta(s))}|$$

This lifting allows us to define an approximate bisimulation relation:

Definition 4.2.2 (Accumulative bisimulation)**.** *A relation $\mathcal{R} \subseteq S \times S \times [0, \epsilon]$ is an* ϵ-accumulative bisimulation *if for all $(s, t, c) \in \mathcal{R}$:*

1. *$s \xrightarrow{a} \mu$ implies $t \xrightarrow{a} \mu'$ and $\mu \mathcal{L}^T(\mathcal{R}, c) \mu'$*

2. $t \xrightarrow{a} \mu'$ *implies* $s \xrightarrow{a} \mu$ *and* $\mu \mathcal{L}^T(\mathcal{R}, c)\mu'$

We denote by $s \prec_T^{(\epsilon,c)} t$, where $c \in [0, \epsilon]$, if there exists an ϵ-accumulative bisimulation $\mathcal{R}$ such that $(s, t, c) \in \mathcal{R}$.

The following proposition lists the metrical properties of ϵ-accumulative bisimulation.

Proposition 4.2.3. *The following hold:*

1. $s \prec_T^{(0,0)} s$;
2. $s_1 \prec_T^{(\epsilon,c)} s_2$ *iff* $s_2 \prec_T^{(\epsilon,c)} s_1$;
3. *If* $s_1 \prec_T^{(\epsilon_1,c_1)} s_2 \prec_T^{(\epsilon_2,c_2)} s_3$ *then* $s_1 \prec_T^{(\epsilon_1+\epsilon_2,c_1+c_2)} s_3$.

Proof. 1. For reflexivity, it is enough to show that the identity relation over the set S of states, that is the relation $Id_S = \{(s, s, 0) | s \in S\}$, is an 0-accumulative bisimulation. This is easy.

2. For symmetry, assume that (s_1, s_2, c) is in an ϵ-accumulative bisimulation $\mathcal{R}$, we will show that $\mathcal{R}' = \{(s_2', s_1', c) | (s_1', s_2', c) \in \mathcal{R}\}$ is an ϵ-accumulative bisimulation, thus we have $s_2 \prec_T^{(\epsilon,c)} s_1$.

   For $(s_2', s_1', c) \in \mathcal{R}'$, if $s_2' \xrightarrow{a} \mu_2$, we must show that there exists a transition from s_1': $s_1' \xrightarrow{a} \mu_1$ and $\mu_2 \mathcal{L}^T(\mathcal{R}', c)\mu_1$. Since $(s_1', s_2', c) \in \mathcal{R}$, there exists a transition from s_1' such that $s_1' \xrightarrow{a} \mu_1$ and $\mu_1 \mathcal{L}^T(\mathcal{R}, c)\mu_2$. According to the definition of T-lifting, there exist a bijection $\beta : supp(\mu_1) \longrightarrow supp(\mu_2)$, such that for all s_1'' in $supp(\mu_1)$, there exists $s_2'' \in supp(\mu_2)$ such that $s_2'' = \beta(s_1'')$ and $(s_1'', s_2'', c + \sigma) \in \mathcal{R}$ where $\sigma = \max_{s_1'' \in supp(\mu_1)} |\ln \frac{\mu_1(s_1'')}{\mu_2(s_2'')}|$. Consider the inverse of the bijection β satisfying $s_1'' = \beta^{-1}(s_2'')$, since

$$\sigma = \max_{s_1'' \in supp(\mu_1)} \big|\ln \frac{\mu_1(s_1'')}{\mu_2(s_2'')}\big| = \max_{s_1'' \in supp(\mu_1)} \big|\ln \frac{\mu_2(s_2'')}{\mu_1(s_1'')}\big|$$

   Then $(s_2'', s_1'', c + \sigma) \in \mathcal{R}'$ and $\mu_2 \mathcal{L}^T(\mathcal{R}', c)\mu_1$ holds.

3. For transitivity, assume that (s_1, s_2, c_1) is in an ϵ_1-accumulative bisimulation $\mathcal{R}_1 \subseteq S \times S \times [0, \epsilon_1]$, (s_2, s_3, c_2) is in an ϵ_2-accumulative bisimulation $\mathcal{R}_2 \subseteq S \times S \times [0, \epsilon_2]$. It suffices to show that their relational composition $\mathcal{R}_1\mathcal{R}_2 \subseteq S \times S \times [0, \epsilon_1 + \epsilon_2]$:

$$\{(s_1', s_3', c) | \exists s_2', c_1, c_2 . (s_1', s_2', c_1) \in \mathcal{R}_1 \wedge (s_2', s_3', c_2) \in \mathcal{R}_2 \wedge c \leq c_1 + c_2\}$$

is an $\epsilon_1 + \epsilon_2$-accumulative bisimulation.

For $(s_1', s_3', c) \in \mathcal{R}_1\mathcal{R}_2$, if $s_1' \xrightarrow{a} \mu_1$, we must show that there exists a transition from s_3': $s_3' \xrightarrow{a} \mu_3$ and $\mu_1 \mathcal{L}^T(\mathcal{R}_1\mathcal{R}_2, c)\mu_3$. Since there exist s_2', c_1, c_2 such that $(s_1', s_2', c_1) \in \mathcal{R}_1$ and $(s_2', s_3', c_2) \in \mathcal{R}_2$ and $c \leq c_1 + c_2$, there exist also a transition $s_2' \xrightarrow{a} \mu_2$ and $\mu_1 \mathcal{L}^T(\mathcal{R}_1, c_1)\mu_2$, and hence a transition $s_3' \xrightarrow{a} \mu_3$ and $\mu_2 \mathcal{L}^T(\mathcal{R}_2, c_2)\mu_3$. By the definition of T-lifting, there exists a bijection $\beta_1 : supp(\mu_1) \longrightarrow supp(\mu_2)$, s.t. for all s_1'' in $supp(\mu_1)$, there exists $s_2'' \in supp(\mu_2)$ such that $s_2'' = \beta_1(s_1'')$ and

$$(s_1'', s_2'', c_1 + \sigma_1) \in \mathcal{R}_1 \text{ where } \sigma_1 = \max_{s_1'' \in supp(\mu_1)} |\ln \frac{\mu_1(s_1'')}{\mu_2(s_2'')}|$$

There exists also a bijection $\beta_2 : supp(\mu_2) \longrightarrow supp(\mu_3)$, s.t. for all s_2'' in $supp(\mu_2)$, there exists $s_3'' \in supp(\mu_3)$ such that $s_3'' = \beta_2(s_2'')$ and

$$(s_2'', s_3'', c_2 + \sigma_2) \in \mathcal{R}_2 \text{ where } \sigma_2 = \max_{s_2'' \in supp(\mu_2)} |\ln \frac{\mu_2(s_2'')}{\mu_3(s_3'')}|$$

Consider the composition $\beta_1\beta_2$ satisfying $\beta_1\beta_2 : supp(\mu_1) \longrightarrow supp(\mu_3)$, s.t. for all s_1'' in $supp(\mu_1)$, there exists $s_3'' \in supp(\mu_3)$ such that $s_3'' = \beta_2(\beta_1(s_1''))$. Let $s_2'' = \beta_1(s_1'')$ and $\sigma' = \max_{s_1'' \in supp(\mu_1)} \big|\ln \frac{\mu_1(s_1'')}{\mu_3(s_3'')}\big|$, we have

$$\begin{aligned}
\sigma' &= \max_{s_1'' \in supp(\mu_1)} \big|\ln \big(\tfrac{\mu_1(s_1'')}{\mu_2(s_2'')} \cdot \tfrac{\mu_2(s_2'')}{\mu_3(s_3'')}\big)\big| \\
&\leq \max_{s_1'' \in supp(\mu_1)} (\big|\ln \tfrac{\mu_1(s_1'')}{\mu_2(s_2'')}\big| + \big|\ln \tfrac{\mu_2(s_2'')}{\mu_3(s_3'')}\big|) \\
&\leq \max_{s_1'' \in supp(\mu_1)} \big|\ln \tfrac{\mu_1(s_1'')}{\mu_2(s_2'')}\big| + \max_{s_2'' \in supp(\mu_1)} \big|\ln \tfrac{\mu_2(s_2'')}{\mu_3(s_3'')}\big| \\
&= \sigma_1 + \sigma_2
\end{aligned}$$

by $c \leq c_1 + c_2$ hence $c + \sigma' \leq c_1 + \sigma_1 + c_2 + \sigma_2$,

$$(s_1'', s_3'', c + \sigma') \in \mathcal{R}_1\mathcal{R}_2$$

and it holds that $\mu_1 \mathcal{L}^T(\mathcal{R}_1 \mathcal{R}_2, c) \mu_3$.

□

Verification of differential privacy using accumulative bisimulation. As shown in [TKD11], the closeness of processes in an indexed family implies a level of differential privacy. We here restate this result in terms of accumulative bisimulation.

Lemma 4.2.4. *Given a PA $\mathcal{A}$, let $\mathcal{R}$ be an ϵ-accumulative bisimulation, $c \in [0, \epsilon]$, let ζ be an admissible scheduler, $\vec{t}$ be a finite trace, α_1, α_2 two finite executions of $\mathcal{A}$ that enjoy the same sequence of labels. If $(lstate(\alpha_1), lstate(\alpha_2), c) \in \mathcal{R}$, then*

$$\frac{1}{e^{\epsilon - c}} \leq \frac{\Pr_\zeta [\alpha_1 \rhd C_{\vec{t}}]}{\Pr_\zeta [\alpha_2 \rhd C_{\vec{t}}]} \leq e^{\epsilon - c}$$

The above lemma shows that in an ϵ-accumulative bisimulation, two states related by a leakage amount c, produce distributions over the same trace that only deviate by a factor $(\epsilon - c)$. Then it is easy to get that the level of differential privacy is continuous with respect to ϵ-accumulative bisimulation.

Theorem 4.2.5. *A concurrent system $\mathcal{A}$ is ϵ-differentially private if*

$$\mathcal{A}(u) \prec_T^{(\epsilon,0)} \mathcal{A}(u') \text{ for all } u \smile u'$$

Note that the converse does not hold. A counter example will be provided in Example 4.3.1 in the next section. It shows a system that is differentially private, however, there does not exists any accumulative bisimulation that can characterize it.

4.3 The amortised bisimulation

As shown in the previous section, accumulative bisimulation is useful for verifying differential privacy. However, a drawback of it is that its definition

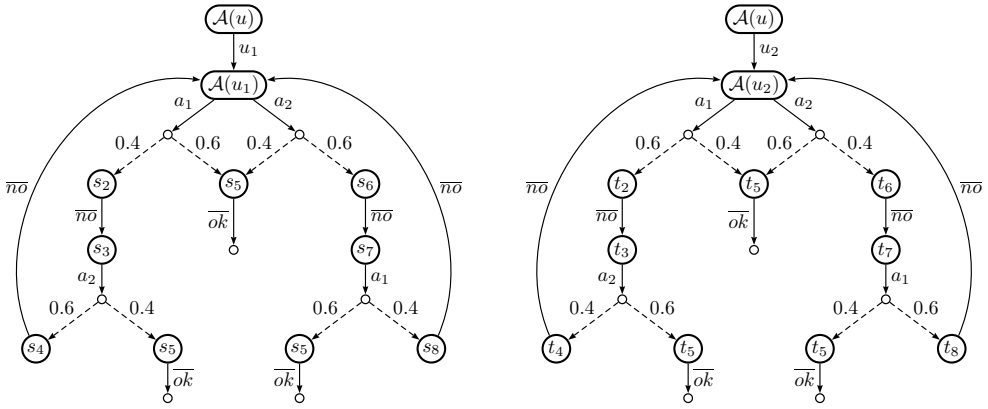

Figure 4.1: A PIN-checking system.

is too restrictive: first, the amount of leakage is only accumulated, independently from whether the difference in probabilities is negative or positive. Moreover, the accumulation is the same for all branches, and equal to the worst branch, although the actual difference on some branch might be small. As a consequence, accumulative bisimulation is inapplicable in several systems, as shown by the following toy example.

Example 4.3.1. *Consider a PIN-checking system $\mathcal{A}(u)$ in which the PIN variable u is chosen from two secret codes u_1 and u_2. In order to protect the secrecy of the two PINs, rather than announcing to a user deterministically whether the password he enters is correct or wrong, the system makes a response probabilistically. The idea is to give a positive answer with a higher probability when the password and the PIN match, and to give a negative answer with a higher probability otherwise.*

The PIN-checking system could be defined as the PA shown in Fig. 4.1[1]*. We use label u_i to represent that the password is u_i, and a_i to model the behavior that the password entered by a user is u_i, where $i \in \{1, 2\}$. We use label $\overline{ok}$ and $\overline{no}$ to represent a positive and a negative answer, respectively.*

[1]The PIN-checking system we show is intentionally designed for showing the difference between the accumulative bisimulation and our bisimulation. Allowing the user to enter u_1 at s_3, t_3 or to enter u_2 at s_7, t_7 may result in some cases that our bisimulation could not characterize either.

Consider an admissible scheduler always choosing for $\mathcal{A}(u_1)$ *the* a_1*-branch (the case for the* a_2*-branch is similar), thus scheduling for* $\mathcal{A}(u_2)$ *also the* a_1*-branch. It is easy to see that the ratio of probabilities for* $\mathcal{A}(u_1)$ *and* $\mathcal{A}(u_2)$ *producing the same finite sequences* $(a_1\overline{no}\,a_2\,\overline{no})^*$ *is* $(\frac{0.4\times0.6}{0.6\times0.4})^* = 1$*. For the remaining sequences* $(a_1\overline{no}\,a_2\,\overline{no})^*a_1\overline{ok}$ *and* $(a_1\overline{no}\,a_2\,\overline{no})^*a_1\overline{no}\,a_2\,\overline{ok}$*, we can check that the ratios are bounded by* $\frac{9}{4}$*. Thus,* $\mathcal{A}$ *satisfies* $\ln\frac{9}{4}$*-differential privacy.*

However, we can not find an accumulative bisimulation with a bounded ϵ *between* $\mathcal{A}(u_1)$ *and* $\mathcal{A}(u_2)$*. The problem lies in that the leakage amount is always accumulated by adding the absolute differences during cyclic simulations, resulting in a convergence to* ∞*.*

Lastly, in accumulative bisimulation, the bijection condition on the support sets of two related distributions is so stringent that bisimilar processes such as $P_1 = \Delta(a.b)$ and $P_2 = 0.5\ a.b \oplus 0.5\ a.(b + \mathbf{0})$ do not satisfy its lifting operation.

In order to obtain a more relaxed bisimulation, we employ the *amortised bisimulation* relation of [KAK05, dFERVGR07]. The main intuition behind this notion is that the privacy leakage amount in each simulation step may be either reduced due to a negative difference of probabilities, or increased due to a positive difference. Hence, the long-term budget gets amortised, in contrast to the accumulative bisimulation in which the budget is always consumed. Note that the current leakage c ranges over $[-\epsilon, \epsilon]$.

Furthermore, to remove the bijection requirement used in the lifting operation of accumulative bisimulation, we use instead two *weight functions*. The following notion of lifting is used to introduce amortised probabilistic bisimulation. Given a relation $\mathcal{R} \subseteq S \times S \times [-\epsilon, \epsilon]$, $\mu\ \mathcal{L}^A(\mathcal{R}, c)\ \mu'$ informally means that there is a way to split the probabilities of the states of μ between the states of μ' and vice versa, expressed by two weight functions ω and ω', so that the relation $\mathcal{R}$ is preserved. In other words, each probability distribution can be embedded to the other up to $\mathcal{R}$ and the given variation c. Using weight functions to define probabilistic relations is not new in the literature, see for instance the notion of simulation defined in [SL95] where

the two weight functions coincide.

Letter A (short for amortisation) is simply used to differentiate the lifting notion of amortised bisimulation from the accumulative one of Tschantz *et al.*.

Definition 4.3.2. *Let* $\epsilon \geqslant 0$, $c \in [-\epsilon, \epsilon]$, $\mathcal{R} \subseteq S \times S \times [-\epsilon, \epsilon]$. *The* A-lifting *of* $\mathcal{R}$ *up to* c, *denoted by* $\mathcal{L}^A(\mathcal{R}, c)$, *is a relation on* $Disc(S)$ *defined as* $\mu\ \mathcal{L}^A(\mathcal{R}, c)\ \mu'$, *iff there exist two weight functions* $\omega, \omega' : S \times S \to [0, 1]$ *such that*

1. *for all* $s \in S$, $\sum_{t \in S} \omega(s, t) = \mu(s)$;
2. *for all* $t \in S$, $\sum_{s \in S} \omega'(s, t) = \mu'(t)$;
3. *for all* $s, t \in S$, $\omega(s, t) = 0$ *iff* $\omega'(s, t) = 0$;
4. *for all* $s, t \in S$, *if* $\omega(s, t) > 0$, *then*

$$(s, t, c + \ln \frac{\omega(s, t)}{\omega'(s, t)}) \in \mathcal{R}.$$

Note that if $\ln \frac{\omega(s,t)}{\omega'(s,t)}$ is positive, then after this mutual step, the amount of deviation between s and t will be increased, otherwise decreased.

Definition 4.3.3 (Amortised bisimulation)**.** *A relation* $\mathcal{R} \subseteq S \times S \times [-\epsilon, \epsilon]$ *is an* ϵ-amortised bisimulation *if for all* $(s, t, c) \in \mathcal{R}$*:*

1. $s \xrightarrow{a} \mu$ *implies* $t \xrightarrow{a} \mu'$ *and* $\mu \mathcal{L}^A(\mathcal{R}, c) \mu'$
2. $t \xrightarrow{a} \mu'$ *implies* $s \xrightarrow{a} \mu$ *and* $\mu \mathcal{L}^A(\mathcal{R}, c) \mu'$

We denote by $Q \prec^{(\epsilon,c)} Q'$, where $|c| \leq \epsilon$, if there exists an ϵ-amortised bisimulation $\mathcal{R}$ such that $(Q, Q', c) \in \mathcal{R}$.

Using $\ln \frac{\omega(s,t)}{\omega'(s,t)}$ in the lifting operation allows us to learn the degree of the behavioral similarity between two related processes, which is, specifically, measured by the ratio between the probabilities of producing the same trace starting from the two processes. The behavioral similarity measured in the multiplicative form is exactly the similarity required by the differential privacy property. This fact will be proved in Lemma 4.3.7 and Theorem 4.3.8.

Example 4.3.4. *Consider again the two bisimilar probabilistic processes* $P_1 = \Delta(a.b)$ *and* $P_2 = 0.5\ a.b \oplus 0.5\ a.(b+\mathbf{0})$ *which accumulative bisimulation fails to equate.* P_1 *has just one successor state* $a.b$ *while* P_2 *has two:* $a.b$ *and* $a.(b+\mathbf{0})$*. For them to match each other, we need to split the probability of* $a.b$ *in* P_1 *into two parts and allocate them to the two states of* P_2*. We can find weight functions* ω, ω' *between them below and conclude that there is a* 0*-amortised bisimulation between* P_1 *and* P_2*.*

$\omega = \omega'$	$a.b$	$a.(b+\mathbf{0})$
$a.b$	*0.5*	*0.5*

In the above example, the split of the probability of a state is needed. The next example shows the case where it happens to be no split of the probability of a state. More precisely, for all $s, t \in S$, $\omega(s,t) = \mu(s)$, if $s = t$, otherwise 0; $\omega'(s,t) = \mu'(t)$, if $s = t$, otherwise 0.

Example 4.3.5. *Consider two processes* E *and* F*:*

$$
\begin{array}{lll}
E & = a.P_1 + b.P_2 & F = a.P_3 + b.P_4 \\
P_1 & = 0.5\ \mathbf{0} \oplus 0.3E_1 \oplus 0.2E_2 & P_3 = 0.5\ \mathbf{0} \oplus 0.2F_1 \oplus 0.3F_2 \\
E_1 & = c.P_4 & F_1 = c.P_2 \\
E_2 & = d.P_5 & F_2 = d.P_2 \\
P_2 & = 0.5ok \oplus 0.5no & P_4 = 0.4ok \oplus 0.6no \\
P_5 & = 0.6ok \oplus 0.4no &
\end{array}
$$

where the Dirac measure $\Delta(\mathbf{0})$ *after* ok *and* no *is omitted for simplicity.*

Take P_2 *and* P_4 *as an example, it is easy to find an* ϵ*-amortised bisimulation* $\mathcal{R}$ *such that*

$$\mathcal{R} = \left\{(P_2, P_4, 0), (ok, ok, \ln\frac{5}{4}), (no, no, \ln\frac{5}{6})\right\}$$

then $\epsilon = \max\{|\ln\frac{5}{4}|, |\ln\frac{5}{6}|\} = \ln\frac{5}{4}$*, and hence* $P_2 \prec^{(\ln\frac{5}{4}, 0)} P_4$*.*

Consider the relation between E *and* F*. The probabilistic automata for* E *and* F *are shown in Figure 4.2. When* P_1 *and* P_2 *are lifted, we can*

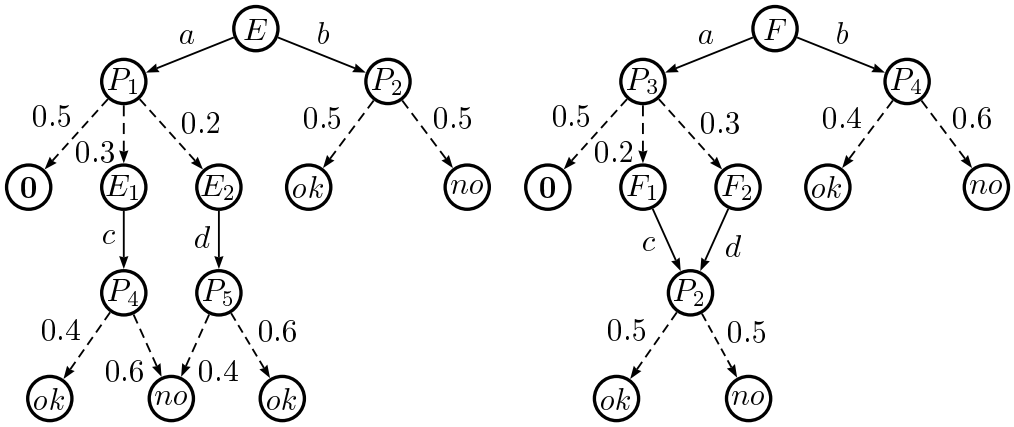

Figure 4.2: The probabilistic automata of Example 4.3.5.

allocate all the probability of reaching E_1 to F_1 and vice versa, and do the same thing between E_2 and F_2. According to Def. 4.3.3, we can check that the following relation $\mathcal{R}$ is a $\ln\frac{9}{5}$-amortised bisimulation between E and F.

$$\begin{aligned}\mathcal{R} = \{\ & (E, F, 0),\\ & (E_1, F_1, \ln\tfrac{3}{2}), (ok, ok, \ln\tfrac{6}{5}), (no, no, \ln\tfrac{9}{5}),\\ & (E_2, F_2, \ln\tfrac{2}{3}), (ok, ok, \ln\tfrac{4}{5}), (no, no, \ln\tfrac{8}{15}),\\ & (\mathbf{0}, \mathbf{0}, 0), (ok, ok, \ln\tfrac{5}{4}), (no, no, \ln\tfrac{5}{6})\ \}.\end{aligned}$$

Analogous to the accumulative bisimulation, the amortised bisimulation enjoys the metrical properties as well.

Proposition 4.3.6. *The following hold:*

1. $s \prec^{(0,0)} s$;
2. $s_1 \prec^{(\epsilon,c)} s_2$ *iff* $s_2 \prec^{(\epsilon,-c)} s_1$;
3. *If* $s_1 \prec^{(\epsilon_1,c_1)} s_2 \prec^{(\epsilon_2,c_2)} s_3$ *then* $s_1 \prec^{(\epsilon_1+\epsilon_2,c_1+c_2)} s_3$.

Proof. 1. For reflexivity, it is enough to show that the identity relation over the set S of states, that is the relation $Id_S = \{(s, s, 0) | s \in S\}$, is an 0-amortised bisimulation. If $s \xrightarrow{a} \mu$, the weight functions ω and

ω' can be defined as: $\omega(s,t) = \omega'(s,t) = \mu(s)$, if $s = t$; otherwise 0. This is easy.

2. For symmetry, assume that (s_1, s_2, c) is in an ϵ-amortised bisimulation $\mathcal{R}$, we will show that $\mathcal{R}' = \{(s_2', s_1', c) | (s_1', s_2', -c) \in \mathcal{R}\}$ is an ϵ-amortised bisimulation, thus we have $s_2 \prec^{(\epsilon,-c)} s_1$.

   For $(s_2', s_1', c) \in \mathcal{R}'$, if $s_2' \xrightarrow{a} \mu_2$, we must show that there exists a transition from s_1': $s_1' \xrightarrow{a} \mu_1$ and $\mu_2 \mathcal{L}^A(\mathcal{R}', c)\mu_1$. Since $(s_1', s_2', -c) \in \mathcal{R}$, there exists a transition from s_1' such that $s_1' \xrightarrow{a} \mu_1$ and $\mu_1 \mathcal{L}^A(\mathcal{R}, -c)\mu_2$. According to the definition of A-lifting, there are two weight functions ω, ω' such that for all $s, t \in S$, $\sum_t \omega(s,t) = \mu_1(s)$, $\sum_s \omega'(s,t) = \mu_2(t)$; $\omega(s,t) = 0$ iff $\omega'(s,t) = 0$; and if $\omega(s,t) > 0$,

   $$(s, t, -c + \ln \frac{\omega(s,t)}{\omega'(s,t)}) \in \mathcal{R}$$

   Straightforwardly, $\mu_2 \mathcal{L}^A(\mathcal{R}', c)\mu_1$ holds, because we can exchange the two weight functions such that $(t, s, c + \ln \frac{\omega'(s,t)}{\omega(s,t)}) \in \mathcal{R}'$.

3. For transitivity, let (s_1, s_2, c_1) be in an ϵ_1-amortised bisimulation $\mathcal{R}_1 \subseteq S \times S \times [-\epsilon_1, \epsilon_1]$, (s_2, s_3, c_2) be in an ϵ_2-amortised bisimulation $\mathcal{R}_2 \subseteq S \times S \times [-\epsilon_2, \epsilon_2]$. Let $\mathcal{R} \subseteq S \times S \times [-\epsilon_1 - \epsilon_2, \epsilon_1 + \epsilon_2]$:

   $$\{\ (s_1, s_3, c)\ | \exists s_2, c_1, c_2 . (s_1, s_2, c_1) \in \mathcal{R}_1 \wedge (s_2, s_3, c_2) \in \mathcal{R}_2 \wedge c = c_1 + c_2\ \}.$$

   We extend $\mathcal{R}$ to be $\mathcal{R}_{\text{Ext}}$ as follows:

   - $\mathcal{R} \subseteq \mathcal{R}_{\text{Ext}}$.
   - If $(s, s', c_1) \in \mathcal{R}_{\text{Ext}}$ and $(s, s', c_2) \in \mathcal{R}_{\text{Ext}}$, then for any c, if $c_1 \leq c \leq c_2$, $(s, s', c) \in \mathcal{R}_{\text{Ext}}$.

   We shall prove that $\mathcal{R}_{\text{Ext}}$ is an $\epsilon_1 + \epsilon_2$-amortised bisimulation.

   Given $(s_1, s_3, c) \in \mathcal{R}_{\text{Ext}}$, there are two cases:

   a) $(s_1, s_3, c) \in \mathcal{R}$. If $s_1 \xrightarrow{a} \mu_1$, by $(s_1, s_2, c_1) \in \mathcal{R}_1$, there exist $s_2 \xrightarrow{a} \mu_2$ and $\mu_1\ \mathcal{L}(\mathcal{R}_1, c_1)\ \mu_2$; by $(s_2, s_3, c_2) \in \mathcal{R}_2$, there exist

$s_3 \xrightarrow{a} \mu_3$ and $\mu_2\ \mathcal{L}(\mathcal{R}_2, c_2)\ \mu_3$. Let ω and ω' be two weight functions between μ_1 and μ_2, γ and γ' be two weight functions between μ_2 and μ_3. We shall construct two weight functions π, π' between μ_1 and μ_3 out of ω, ω', γ and γ' in such a way that $\mu_1\ \mathcal{L}(\mathcal{R}_{\text{Ext}}, c)\ \mu_3$ where $c = c_1 + c_2$.

Define weight functions $\pi, \pi' : S \times S \to [0, 1]$ as follows:

$$\pi(s,t) = \sum_r \frac{\omega(s,r)\gamma(r,t)}{\mu_2(r)},$$

$$\pi'(s,t) = \sum_r \frac{\omega'(s,r)\gamma'(r,t)}{\mu_2(r)}.$$

We can check that $\sum_t \pi(s,t) = \mu_1(s)$ and $\sum_s \pi'(s,t) = \mu_3(t)$; and $\pi(s,t) = 0$ iff $\pi'(s,t) = 0$. If $\pi'(s,t) \neq 0$, since $\min_r \ln \frac{\omega(s,r)\gamma(r,t)}{\omega'(s,r)\gamma'(r,t)} \leq \ln \frac{\pi(s,t)}{\pi'(s,t)} \leq \max_r \ln \frac{\omega(s,r)\gamma(r,t)}{\omega'(s,r)\gamma'(r,t)}$,

$$(s, t, c + \min_r \ln \frac{\omega(s,r)\gamma(r,t)}{\omega'(s,r)\gamma'(r,t)}) \in \mathcal{R},$$

and

$$(s, t, c + \max_r \ln \frac{\omega(s,r)\gamma(r,t)}{\omega'(s,r)\gamma'(r,t)}) \in \mathcal{R},$$

by the definition of $\mathcal{R}_{\text{Ext}}$,

$$(s, t, c + \ln \frac{\pi(s,t)}{\pi'(s,t)}) \in \mathcal{R}_{\text{Ext}}.$$

Thus π, π' are the two required weight functions and $\mu_1\ \mathcal{L}(\mathcal{R}_{\text{Ext}}, c)\ \mu_3$ holds.

b) Otherwise, $(s_1, s_3, c) \in \mathcal{R}_{\text{Ext}} \backslash \mathcal{R}$. Namely, there exist c_1 and c_2, $c_1 \leq c \leq c_2$, $(s_1, s_3, c_1) \in \mathcal{R}$ and $(s_1, s_3, c_2) \in \mathcal{R}$.

If $s_1 \xrightarrow{a} \mu$, by $(s_1, s_3, c_1) \in \mathcal{R}$ and the result of case 3a, there exists $s_3 \xrightarrow{a} \mu'$ and $\mu \mathcal{L}(\mathcal{R}_{\text{Ext}}, c_1) \mu'$. Namely there are two weight functions ω_1 and ω_1' s.t. $\sum_t \omega_1(s,t) = \mu(s)$, $\sum_s \omega_1'(s,t) = \mu'(t)$ and

$$(s, t, c_1 + \ln \frac{\omega_1(s,t)}{\omega_1'(s,t)}) \in \mathcal{R}_{\text{Ext}}.$$

Analogously, by $(s_1, s_3, c_2) \in \mathcal{R}$, there are another two weight functions ω_2 and ω_2' s.t. $\sum_t \omega_2(s,t) = \mu(s)$, $\sum_s \omega_2'(s,t) = \mu'(t)$ and

$$(s, t, c_2 + \ln \frac{\omega_2(s,t)}{\omega_2'(s,t)}) \in \mathcal{R}_{\text{Ext}}.$$

We define weight functions ω and ω' s.t. $\omega(s,t) = \frac{1}{2}\omega_1(s,t) + \frac{1}{2}\omega_2(s,t)$ and $\omega'(s,t) = \frac{1}{2}\omega_1'(s,t) + \frac{1}{2}\omega_2'(s,t)$. By the definition of $\mathcal{R}_{\text{Ext}}$, it holds

$$(s, t, c + \ln \frac{\omega(s,t)}{\omega'(s,t)}) \in \mathcal{R}_{\text{Ext}}.$$

Hence, $\mathcal{R}_{\text{Ext}}$ is an $\epsilon_1 + \epsilon_2$-amortised bisimulation.

□

Verification of differential privacy using amortised bisimulation. We now show that amortised bisimulation can be used to verify differential privacy.

Lemma 4.3.7. *Given a PA $\mathcal{A}$, let $\mathcal{R}$ be an ϵ-amortised bisimulation, $c \in [-\epsilon, \epsilon]$, let ζ be an admissible scheduler, $\vec{t}$ be a finite trace, α_1, α_2 two finite executions of $\mathcal{A}$ that enjoy the same sequence of labels.*

If $(lstate(\alpha_1), lstate(\alpha_2), c) \in \mathcal{R}$, then

$$\frac{1}{e^{\epsilon+c}} \leq \frac{\Pr_\zeta[\alpha_1 \rhd C_{\vec{t}}]}{\Pr_\zeta[\alpha_2 \rhd C_{\vec{t}}]} \leq e^{\epsilon-c}$$

Proof. We prove by induction on the length of trace $\vec{t}$: $|\vec{t}|$.

1. $|\vec{t}| = 0$: According to Eq. (4.1), for any ζ,

$$\Pr_\zeta[\alpha_1 \rhd C_{\vec{t}}] = \Pr_\zeta[\alpha_2 \rhd C_{\vec{t}}] = 1$$

2. IH: For any two executions α_1 and α_2 of $\mathcal{A}$, let $s_1 = lstate(\alpha_1)$ and $s_2 = lstate(\alpha_2)$. $(s_1, s_2, c) \in \mathcal{R}$ implies that for any admissible scheduler ζ, trace $\vec{t'}$ where $|\vec{t'}| \leq L$:

$$\frac{1}{e^{\epsilon+c}} \leq \frac{\Pr_\zeta[\alpha_1 \rhd C_{\vec{t'}}]}{\Pr_\zeta[\alpha_2 \rhd C_{\vec{t'}}]} \leq e^{\epsilon-c}$$

3. We have to show that for any admissible scheduler ζ, trace $\vec{t}$ with $|\vec{t}| = L + 1$, $(s_1, s_2, c) \in \mathcal{R}$ implies

$$\frac{1}{e^{\epsilon+c}} \leq \frac{\Pr_\zeta [\alpha_1 \rhd C_{\vec{t}}\,]}{\Pr_\zeta [\alpha_2 \rhd C_{\vec{t}}\,]} \leq e^{\epsilon-c}$$

Assume that $\vec{t} = a^\frown \vec{t'}$. We prove first the right-hand part $\Pr_\zeta [\alpha_1 \rhd C_{\vec{t}}\,] \leq e^{\epsilon-c} \cdot \Pr_\zeta [\alpha_2 \rhd C_{\vec{t}}\,]$. According to Eq. (4.1), two cases must be considered:

- Case $act(\zeta(\alpha_1)) \neq a$. Then $\Pr_\zeta [\alpha_1 \rhd C_{\vec{t}}\,] = 0$. Since ζ is admissible, it schedules for α_2 a transition consistent with α_1, namely, not a transition labeled by a either. Thus $\Pr_\zeta [\alpha_2 \rhd C_{\vec{t}}\,] = 0$, the inequality is satisfied.
- Case $\zeta(\alpha_1) = s_1 \xrightarrow{a} \mu_1$. So,

$$\Pr_\zeta [\alpha_1 \rhd C_{\vec{t}}\,] = \textstyle\sum_s \mu_1(s) \Pr_\zeta [\alpha_1 a s \rhd C_{\vec{t'}}\,] \tag{4.2}$$

Since $(s_1, s_2, c) \in \mathcal{R}$, there must be also a transition from s_2 such that $s_2 \xrightarrow{a} \mu_2$ and $\mu_1 \mathcal{L}^A(\mathcal{R}, c) \mu_2$. Since ζ is admissible, $\zeta(\alpha_2) = s_2 \xrightarrow{a} \mu_2$. Thus,

$$\Pr_\zeta [\alpha_2 \rhd C_{\vec{t}}\,] = \textstyle\sum_t \mu_2(t) \cdot \Pr_\zeta [\alpha_2 a t \rhd C_{\vec{t'}}\,] \tag{4.3}$$

Since $\mu_1 \mathcal{L}^A(\mathcal{R}, c) \mu_2$, there are two weight functions ω, ω' such that for all $s, t \in S$, $\sum_t \omega(s,t) = \mu_1(s)$, $\sum_s \omega'(s,t) = \mu_2(t)$; $\omega(s,t) = 0$ iff $\omega'(s,t) = 0$; and if $\omega(s,t) > 0$,

$$(s, t, c + \ln \frac{\omega(s,t)}{\omega'(s,t)}) \in \mathcal{R}$$

Apply the inductive hypothesis to $\alpha_1 a s, \alpha_2 a t$ and $\vec{t'}$, we get that:

$$\Pr_\zeta [\alpha_1 a s \rhd C_{\vec{t'}}\,] \leq e^{\epsilon - (c + \ln \frac{\omega(s,t)}{\omega'(s,t)})} \cdot \Pr_\zeta [\alpha_2 a t \rhd C_{\vec{t'}}\,] \tag{4.4}$$

Therefore,

$$
\begin{aligned}
& \Pr_\zeta [\alpha_1 \rhd C_{\vec{t}}\,] && (4.5)\\
= & \sum_s \mu_1(s) \Pr_\zeta [\alpha_1 a s \rhd C_{\vec{t'}}\,] && (4.6)\\
= & \sum_s \sum_t \omega(s,t) \Pr_\zeta [\alpha_1 a s \rhd C_{\vec{t'}}\,] && (4.7)\\
\leq & \sum_{s,t} \omega(s,t) e^{\epsilon - (c + \ln \frac{\omega(s,t)}{\omega'(s,t)})} \Pr_\zeta [\alpha_2 a t \rhd C_{\vec{t'}}\,] && (4.8)\\
= & \sum_{s,t} \omega(s,t) \cdot \frac{\omega'(s,t)}{\omega(s,t)} \cdot e^{\epsilon - c} \cdot \Pr_\zeta [\alpha_2 a t \rhd C_{\vec{t'}}\,] && (4.9)\\
= & \sum_t \sum_s \omega'(s,t) \cdot e^{\epsilon - c} \cdot \Pr_\zeta [\alpha_2 a t \rhd C_{\vec{t'}}\,] && (4.10)\\
= & \sum_t \mu_2(t) \cdot e^{\epsilon - c} \cdot \Pr_\zeta [\alpha_2 a t \rhd C_{\vec{t'}}\,] && (4.11)\\
= & e^{\epsilon - c} \cdot \sum_t \mu_2(t) \Pr_\zeta [\alpha_2 a t \rhd C_{\vec{t'}}\,] && (4.12)\\
= & e^{\epsilon - c} \cdot \Pr_\zeta [\alpha_2 \rhd C_{\vec{t}}\,] && (4.13)
\end{aligned}
$$

which completes the proof of the right-hand part. Lines (4.6) and (4.13) follow from the equations (4.2) and (4.3) respectively, Lines (4.7) and (4.11) from the definitions of ω and ω' respectively, Line (4.8) from the inductive hypothesis Line (4.4).

For the left-hand part $\Pr_\zeta [\alpha_2 \rhd C_{\vec{t}}] \leq e^{\epsilon + c} \cdot \Pr_\zeta [\alpha_1 \rhd C_{\vec{t}}]$, exchange the roles of s_1 and s_2, ω and ω', and all the rest is analogous.

□

Note that there is a subtle difference between Lemmas 4.2.4 and 4.3.7, in that the denominator in the left-hand bound is $e^{\epsilon+c}$ instead of $e^{\epsilon-c}$. This comes from the amortised nature of $\mathcal{R}$. We can now show that differential privacy is continuous with respect to ϵ-amortised bisimulation as well.

Theorem 4.3.8. *A concurrent system $\mathcal{A}$ is ϵ-differentially private if*

$$\mathcal{A}(u) \prec^{(\epsilon,0)} \mathcal{A}(u') \text{ for all } u \smile u'$$

Proof. For all $u \backsim u'$, there exists an ϵ-amortised bisimulation $\mathcal{R}$ such that $(\mathcal{A}(u), \mathcal{A}(u'), 0) \in \mathcal{R}$. By Lemma 4.3.7, for any admissible scheduler ζ, any finite trace $\vec{t}$:

$$\frac{1}{e^\epsilon} \leq \frac{\Pr_\zeta\left[\mathcal{A}(u) \rhd C_{\vec{t}}\right]}{\Pr_\zeta\left[\mathcal{A}(u') \rhd C_{\vec{t}}\right]} \leq e^\epsilon$$

Thus, $\mathcal{A}$ is ϵ-differentially private. □

Example 4.3.9 (Example 4.3.1 revisited)**.** *Consider again the concurrent system shown in Fig. 4.1. Let S and T denote the state space of $\mathcal{A}(u_1)$ and $\mathcal{A}(u_2)$, respectively. Let $\mathcal{R} \subseteq S \times T \times [\ln\frac{4}{9}, \ln\frac{9}{4}]$. It is straightforward to check according to Def. 4.3.3 that the following relation is a $\ln\frac{9}{4}$-amortised bisimulation between $\mathcal{A}(u_1)$ and $\mathcal{A}(u_2)$.*

$$\begin{aligned}\mathcal{R} \quad = \{ \quad & (\mathcal{A}(u_1), \mathcal{A}(u_2), 0), \\ & (s_2, t_2, \ln\tfrac{2}{3}), (s_5, t_5, \ln\tfrac{3}{2}), (s_3, t_3, \ln\tfrac{2}{3}), (s_4, t_4, 0), (s_5, t_5, \ln\tfrac{4}{9}), \\ & (s_6, t_6, \ln\tfrac{3}{2}), (s_5, t_5, \ln\tfrac{2}{3}), (s_7, t_7, \ln\tfrac{3}{2}), (s_8, t_8, 0), (s_5, t_5, \ln\tfrac{9}{4}) \}\end{aligned}$$

By Theorem 4.3.8, $\mathcal{A}$ is $\ln\frac{9}{4}$-differentially private.

4.4 Comparing the two bisimulations

In this section, we formally compare the two bisimulations, showing that our amortised bisimulation is indeed more liberal than the accumulative bisimulation. Moreover, we show that 0-accumulative bisimulation only implies bisimilarity, but the converse direction does not hold because of the strong requirement of the bijection in their definition; however our 0-amortised bisimulation can fully characterize bisimilarity.

We show that amortised bisimilarity $\prec$ is more liberal than accumulative bisimilarity $\prec_T$. The converse does not hold, since Examples 4.3.1, 4.3.4 and 4.3.9 already show the cases in which ϵ-accumulative bisimulation is infinite while ϵ-amortised bisimulation is finite.

Lemma 4.4.1. *$s \prec_T^{(\epsilon, c^T)} t$ implies $s \prec^{(\epsilon, c^A)} t$ where $|c^A| \leq c^T$.*

Proof. Assume that $\mathcal{R}^T \subseteq S \times S \times [0, \epsilon]$ is an ϵ-accumulative bisimulation such that $(s, t, c^T) \in \mathcal{R}^T$. We define a relation $\mathcal{R}^A \subseteq S \times S \times [-\epsilon, \epsilon]$ out of $\mathcal{R}^T$ as follows:

$$(s', t', c^A) \in \mathcal{R}^A \text{ iff } \exists c^T.(s', t', c^T) \in \mathcal{R}^T \wedge |c^A| \leq c^T \tag{4.14}$$

Given $(s', t', c^A) \in \mathcal{R}^A$, if $s' \xrightarrow{a} \mu_1$, we must show that there exists a transition from t': $t' \xrightarrow{a} \mu_2$ and $\mu_1 \mathcal{L}^A(\mathcal{R}^A, c^A)\mu_2$. By Eq. (4.14) we know that there exists c^T such that $|c^A| \leq c^T$ and $(s', t', c^T) \in \mathcal{R}^T$. Thus there exists a transition from t' such that $t' \xrightarrow{a} \mu_2$ and $\mu_1 \mathcal{L}^T(\mathcal{R}^T, c^T)\mu_2$. According to the definition of T-lifting, there exists a bijection $\beta : supp(\mu_1) \longrightarrow supp(\mu_2)$, s.t. for all s'' in $supp(\mu_1)$, there exists $t'' \in supp(\mu_2)$, $t'' = \beta(s'')$, $(s'', t'', c^T + \sigma) \in \mathcal{R}^T$ where $\sigma = \max_{s'' \in supp(\mu_1)} |\ln \frac{\mu_1(s'')}{\mu_2(t'')}|$. We have $|c^A + \ln \mu_1(s'') - \ln \mu_2(t'')| \leq c^T + \sigma$ and hence $(s'', t'', c^A + \ln \mu_1(s'') - \ln \mu_2(t'')) \in \mathcal{R}^A$ by Eq. (4.14).

Define the two weight function ω, ω' between μ_1 and μ_2 as follows:

$$\omega(s,t) = \begin{cases} \mu_1(s) & \text{if } t = \beta(s); \\ 0 & \text{otherwise.} \end{cases}$$

$$\omega'(s,t) = \begin{cases} \mu_2(t) & \text{if } t = \beta(s); \\ 0 & \text{otherwise.} \end{cases}$$

According to Definition 4.3.2 of A-lifting, it holds that $\mu_1 \mathcal{L}^A(\mathcal{R}^A, c^A)\mu_2$ as required. □

4.4.1 Relations with conventional probabilistic bisimilarity

The following propositions say that the 0-accumulative bisimulation implies conventional probabilistic bisimilarity, while 0-amortised bisimulation fully characterizes bisimilarity.

Proposition 4.4.2. $s \prec_T^{(0,0)} t \Rightarrow s \sim t$.

Proof. Consider the relation $\mathcal{R}$ induced by 0-accumulative bisimulation. Namely,

$$(s,t) \in \mathcal{R} \text{ iff } s \prec_T^{(0,0)} t.$$

Clearly it is an equivalence relation. We show that it is a probabilistic bisimulation. Let $s \prec_T^{(0,0)} t$. Consider some $s \xrightarrow{a} \mu_1$. Since $s \prec_T^{(0,0)} t$, there exists an 0-amortised bisimulation $\mathcal{R}' \subseteq S \times S \times [0,0]$ such that $(s,t,0) \in \mathcal{R}'$. There exist a bijection β and a distribution μ_2 such that $t \xrightarrow{a} \mu_2$, for any $s_i \in supp(\mu_1)$, there exists $t_i \in supp(\mu_2)$, $t_i = \beta(s_i)$ and $(s_i, t_i, \sigma) \in \mathcal{R}'$ where $\sigma = \max_s |\ln \frac{\mu(s)}{\mu'(\beta(s))}|$. Because the leakage budget is 0, which says that during the mutual simulation, every step must have exactly the same probability, i.e. $\mu_1(s_i) = \mu_2(t_i)$. Furthermore by $(s_i, t_i, 0) \in \mathcal{R}'$, we have $s_i \prec_T^{(0,0)} t_i$, thus $[s_i] = [t_i]$. Henceforth, $\mu_1([s_i]) = \mu_2([s_i])$ for all $[s_i] \in S/\mathcal{R}$ as required. □

Proposition 4.4.3. $s \prec^{(0,0)} t \Leftrightarrow s \sim t$.

Proof. ($\subseteq$) Consider the relation $\mathcal{R}$ induced by 0-amortised bisimulation. Namely,

$$(s,t) \in \mathcal{R} \text{ iff } s \prec^{(0,0)} t.$$

Clearly, $\mathcal{R}$ is an equivalence relation. The procedure of showing that it is a probabilistic bisimulation proceeds analogously to the proof in Proposition 4.4.2.

($\supseteq$) Let

$$\mathcal{R} = \{(s,t,0) \mid s \sim t\}.$$

We need to prove that $\mathcal{R}$ is a 0-amortised bisimulation. The key is to show that for any $\mu_1, \mu_2 \in Disc(S)$:

$$\mu_1 \mathcal{L}(\sim) \mu_2 \text{ implies } \mu_1 \mathcal{L}^A(\mathcal{R}, 0) \mu_2.$$

By Lemma 18 in [SL95], $\mu_1 \mathcal{L}(\sim) \mu_2$ implies that there exists a weight function ω such that $\sum_t \omega(s,t) = \mu_1(s)$, $\sum_s \omega(s,t) = \mu_2(t)$ and if $\omega(s,t) > 0$ then $s \sim t$, namely, $(s,t,0) \in \mathcal{R}$. Thus $\mu_1 \mathcal{L}^A(\mathcal{R}, 0) \mu_2$ holds as required. □

4.5 Congruence

In this section we consider a simple process calculus. It contains prefixing, non-deterministic choice, probabilistic choice, restriction and parallel composition constructors and show that ϵ-amortised bisimulation is substitutive under them. The syntax and semantics of the process calculus have already been introduced in Section 2.3 and Fig. 2.1, respectively.

Proposition 4.5.1. *If* $Q \prec^{(\epsilon,c)} Q'$, *then*

1. $a.Q \prec^{(\epsilon,c)} a.Q'$
2. $R \oplus_p Q \prec^{(\epsilon,c)} R \oplus_p Q'$
3. $R + Q \prec^{(\epsilon,c)} R + Q'$
4. $(\nu a)Q \prec^{(\epsilon,c)} (\nu a)Q'$
5. $R \,|\, Q \prec^{(\epsilon,c)} R \,|\, Q'$.

Proof sketch. Let $\mathcal{R}$ be an ϵ-amortised bisimulation with $(Q, Q', c) \in \mathcal{R}$. Define the relation $Id_S = \{(s, s, 0) | s \in S\}$. We construct for each clause a relation $\mathcal{R}'$ as follows and show that it is an ϵ-amortised bisimulation relation.

1. $\mathcal{R}' = \{\, (a.Q, a.Q', 0) \,\} \cup \mathcal{R}$,
2. $\mathcal{R}' = \{\, (R \oplus_p Q, R \oplus_p Q', 0) \,\} \cup \mathcal{R} \cup Id_R$,
3. $\mathcal{R}' = \{\, (R + Q, R + Q', 0) \,\} \cup \mathcal{R} \cup Id_R$,
4. $\mathcal{R}' = \{\, ((\nu a)Q, (\nu a)Q', c) \mid (Q, Q', c) \in \mathcal{R} \,\}$,
5. $\mathcal{R}' = \{\, (R \,|\, Q, R \,|\, Q', c) \mid (Q, Q', c) \in \mathcal{R} \,\} \cup Id_R$.

We detail the proof for Case *2*, the case when $Q = Q'$ is trivial. Otherwise, let the two weight functions ω, ω' between $R \oplus_p Q$ and $R \oplus_p Q'$ be defined

as follows:

$$\omega(E_1, E_2) = \omega'(E_1, E_2) = \begin{cases} p & \text{if } E_1 = E_2 = R, \\ 1-p & \text{if } E_1 = Q \text{ and } E_2 = Q', \\ 0 & \text{otherwise.} \end{cases}$$

such that $(R, R, 0) \in \mathcal{R}'$ and $(Q, Q', 0) \in \mathcal{R}'$. □

However ϵ-accumulative bisimulation is not substitutive under the probabilistic choice. For example, we have $a.b \prec_T^{(0,0)} a.(b + \mathbf{0})$, while due to the bijection requirement, there is no accumulative bisimulation between $a.b \oplus_{0.5} a.b$ and $a.b \oplus_{0.5} a.(b + \mathbf{0})$.

4.6 An application to the Dining Cryptographers Protocol

In this section we use the bisimulation method to reason about the degree of differential privacy of the Dining Cryptographers Protocol [Cha88] with biased coins. In particular, we show that with probability-p biased coins, the degree of differential privacy in the case of three cryptographers is $|\ln \frac{p}{1-p}|$. This result can also be generalized to the case of n cryptographers.

The problem of the Dining Cryptographers is the following: Three cryptographers dine together. After the dinner, the bill has to be paid by either one of them or by another agent called the master. The master decides who will pay and then informs each of them separately whether he has to pay or not. The cryptographers would like to find out whether the payer is the master or one of them. However, in the latter case, they wish to keep the payer anonymous.

The Dining Cryptographers Protocol (DCP) solves the above problem as follows: each cryptographer tosses a fair coin which is visible to himself and his neighbor to the left. Each cryptographer checks his own coin and the one to his right and, if he is not paying, announces "agree" if the two coins are the same and "disagree" otherwise. However, the paying cryptographer

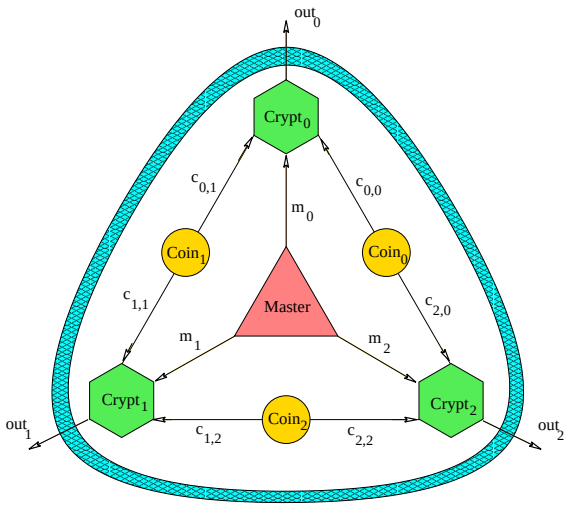

Figure 4.3: Chaum's system for the Dining Cryptographers.

says the opposite. It can be proved that the master is paying if and only if the number of disagrees is even [Cha88].

The graph shown in Fig. 4.3 illustrates the dinner-table and the allocation of the coins between the three cryptographers. We consider the coins which are probability-p biased, i.e., producing 0 (for "head") with probability p, and 1 (for "tail") with $1-p$. We consider the final announcement in the order of $out_0 out_1 out_2$, with $out_i \in \{a, d\}$ (a for "agree" and d for "disagree", $i \in \{0, 1, 2\}$) announced by $Crytp_i$. For example, if $Crytp_0$ is designated to pay, $Coin_0 Coin_1 Coin_2 = 010$, then $out_0 out_1 out_2 = ada$.

We are interested in the case when one of the cryptographers is paying, since that is the case when they want to keep the payer anonymous. We use $Master(m_i)$ to denote the system in which $Crytp_i$ is designated to pay. To show that the DCP is differentially private, both bisimulations introduced before can be used. In this problem, it suffices to find between $Master(m_i)$'s bounded accumulative bisimulation relations.

Proposition 4.6.1. *A DCP with three cryptographers and with probability-p biased coins is $|\ln \frac{p}{1-p}|$-differentially private.*

Proof. Fig. 4.4 shows two probabilistic automata $Master(m_0)$ and $Master(m_1)$ when $Crytp_0$ and $Crytp_1$ are paying respectively. Basically they are proba-

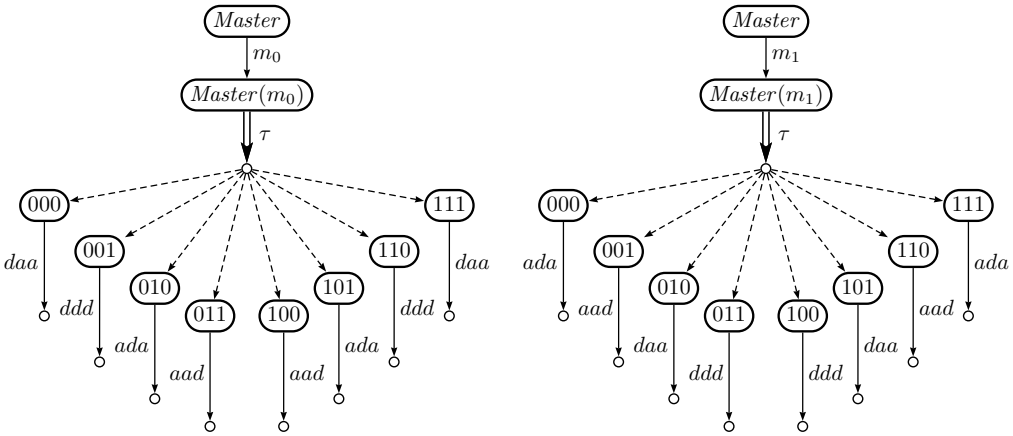

Figure 4.4: The probabilistic automata of the Dining cryptographers.

bilistic distributions over all possible outcomes $Coin_0\,Coin_1\,Coin_2$ (i.e. inner states) produced by the three-coins toss, followed by an announcement determined by each outcome. For simplicity initial τ transitions are merged harmlessly. Let $b_0b_1b_2$ and $c_0c_1c_2$ represent two inner states of $Master(m_0)$ and $Master(m_1)$ respectively. There exists a bijection f between them:

$$c_0c_1c_2 = f(b_0b_1b_2) = b_0(b_1 \oplus 1)b_2$$

where $\oplus$ represents the addition modulo 2 (xor), such that the announcement of $b_0b_1b_2$ can be shown equal to the one of $c_0c_1c_2$.

Note that, the probability of reaching an inner state $b_0b_1b_2$ from $Master(m_0)$ is $p^i(1-p)^{(3-i)}$, where $i \in \{0,1,2,3\}$ is the number of zeroes in $\{b_0, b_1, b_2\}$. Because $c_0 = b_0, c_1 = b_1 \oplus 1, c_2 = b_2$, the ratio between the probabilities of reaching $b_0b_1b_2$ from $Master(m_0)$ and $c_0c_1c_2$ from $Master(m_1)$ differs at most by $|\ln \frac{p}{1-p}|$. It is easy to see that $\{(Master(m_0), Master(m_1), 0)\} \cup \{\, (b_0b_1b_2, f(b_0b_1b_2), |\ln \frac{p}{1-p}|) \mid b_0, b_1, b_2 \in \{0,1\}\,\}$ forms a $|\ln \frac{p}{1-p}|$-accumulative bisimulation relation. Thus $Master(m_0) \prec_T^{(|\ln \frac{p}{1-p}|,0)} Master(m_1)$.

Similarly, we consider the probabilistic automata $Master(m_2)$ when $Crytp_2$ is paying (though omitted in Fig. 4.4). Let $e_0e_1e_2$ represent one of its inner states. We can also find a bijection f' between $c_0c_1c_2$ and $e_0e_1e_2$:

$$e_0e_1e_2 = f'(c_0c_1c_2) = c_0c_1(c_2 \oplus 1)$$

and a bijection f'' between $b_0b_1b_2$ and $e_0e_1e_2$:

$$e_0e_1e_2 = f''(b_0b_1b_2) = (b_0 \oplus 1)b_1b_2$$

such that they output same announcements. The rest proceeds as above. By Theorem 4.2.5, the DCP is $|\ln \frac{p}{1-p}|$-differentially private. □

The above proposition can be extended to the case of n dining cryptographers where $n \geq 3$. We assume that the n cryptographers are fully connected, i.e., that a coin exists between every pair of cryptographers. Let c_{kl} $(k, l \in Z, k, l \in [0, n-1], k < l)$ be the coin linking two cryptographers $Crytp_k$ and $Crytp_l$. In this case the output of $Crytp_i$ would be $out_i = c_{0i} \oplus c_{1i} \oplus \cdots c_{i(n-1)} \oplus pay(i)$, where $pay(i) = 1$ if $Crytp_i$ pays and 0 otherwise.

Proposition 4.6.2. *A DCP with n fully connected cryptographers and with probability-p biased coins is* $|\ln \frac{p}{1-p}|$*-differentially private.*

Proof sketch. The proof proceeds analogously to the case of three cryptographers. To find an accumulative bisimulation relation between every two instances of the DCP $Master(m_i)$ and $Master(m_j)$, $(i, j \in Z, i, j \in [0, n-1], i < j)$, we point out here mainly the bijection between their inner states. Let $b_{12}b_{13}\cdots b_{(n-1)n}$ and $c_{12}c_{13}\cdots c_{(n-1)n}$ represent the inner states of $Master(m_i)$ and $Master(m_j)$ respectively, where the subscript (kl), $(k, l \in Z, k, l \in [0, n-1], k < l)$, indicates the coin linking two cryptographers $Crytp_k$ and $Crytp_l$. There exists a bijection f between them defined as: $c_{12}c_{13}\cdots c_{(n-1)n} = f(b_{12}b_{13}\cdots b_{(n-1)n})$, precisely,

$$c_{kl} = \begin{cases} b_{kl} \oplus 1 & \text{if } kl = ij, \\ b_{kl} & \text{otherwise.} \end{cases}$$

We can check that the bijective states defined in this way produce the same announcement in $Master(m_i)$ and $Master(m_j)$. Moreover, only the coin (ij) is different, the ratio between the probability mass of every pair of bijective states is at most $|\ln \frac{p}{1-p}|$. □

We can see that the more the coins are biased, the worse the privacy gets. If the coins are fair, namely, $p = 1 - p = \frac{1}{2}$, then the DCP is 0-differentially private, in which case the privacy is well protected. With the help of the bisimulation method, we get a general proposition about the degree of differential privacy of DCP. Moreover, it is obtained through some local information, rather than by computing globally the summations of probabilities for each trace.

4.7 Conclusion

In this section we have first studied the metrical properties of the accumulative bisimulation, which is a reformulation of the notion proposed in [TKD11], and proposed the amortised bisimulation where the total privacy leakage gets amortised. Both of them establish a framework for the formal verification of differential privacy for concurrent systems. Namely, the closer processes are in the bisimulations, the higher level of differential privacy they can preserve. We have showed that the amortised bisimulation is more liberal than the former one; it fully characterizes bisimilarity while the accumulative bisimulation only implies bisimilarity; moreover, the amortised bisimulation is substitutive under typical process algebra operators. We have used the bisimulation verification method to learn that: A Dining Cryptographers protocol with probability-p biased coins is $|\ln \frac{p}{1-p}|$-differentially private.

Related Works. Amortised bisimulations were initially proposed in [KAK05] and further studied in [dFERVGR07, Hen11, DH13] concerning cost-based and weight-based quantitative behaviors of processes. In [KAK05, Hen11, DH13] a given budget can be increased or decreased but never gets negative. In our setting the variations are within a given distance but ranging from $-\epsilon$ to ϵ, which is similar to the idea in [dFERVGR07].

A similar notion, called ϵ-bisimulation, was proposed for labelled Markov processes in [DLT08, TDZ11]. Although their ϵ-bisimulation and ϵ-amortised bisimulation in this thesis are both proposed to allow small deviations in

probabilities when comparing processes, our notion is intended to characterize the kind of deviation as defined by ϵ-differential privacy. Because of this motivation, our notion is different from theirs in two aspects: (a) the difference between probabilities is measured in a multiplicative sense (thanks to the ln expression), rather than an additive sense as in [DLT08, TDZ11]; (b) the *total* variation allowed is ϵ, while in [DLT08, TDZ11] the variation allowed in *each simulation step* is ϵ.

Five

Complete Proof Systems for Amortised Probabilistic Bisimulations

The aim of this chapter is to provide sound and complete proof systems for the amortised bisimulation proposed in the previous chapter. Proof systems are important both at the theoretical level, as they provide a deep insight into the nature of process combinators and of the kind of transformations of expressions which preserve bisimulations, and at the practical level, as they provide a foundation for developing verification tools.

As introduced in the previous chapter, for two processes Q and Q', we denote by $Q \prec^{(\epsilon,c)} Q'$ if there exists an ϵ-amortised bisimulation $\mathcal{R}$ such that $(Q, Q', c) \in \mathcal{R}$. The judgments of our inference system are therefore *indexed inequalities* of the form

$$(\epsilon, c) \rhd Q \prec Q'$$

The proof system consists of a set of axioms and a set of inference rules. The axioms include the standard monoid laws for bisimulation and three laws for probabilistic choice. The set of inference rules encompasses rules for manipulating the combinators in the calculus, plus one structural rule used to glue pieces of derivations together.

Since the usual rules for equality (substitutivity, reflexivity, symmetry and transitivity) cannot be straightforwardly applied when reasoning with

inequalities, especially on indexed inequalities, the completeness proof relies on a careful maintenance of indices when rewriting terms. In particular, the set of $(0,0)$-indexed pairs of states is still an equivalence relation, while for non-$(0,0)$-indexed pairs, the inference rule for transitivity takes a form like triangle inequality, more specifically:

$$\textbf{Triangle}\ \frac{(\epsilon_1, c_1) \rhd Q_1 \prec Q_2 \qquad (\epsilon_2, c_2) \rhd Q_2 \prec Q_3}{(\epsilon_1 + \epsilon_2, c_1 + c_2) \rhd Q_1 \prec Q_3}$$

We also developed weak notions of amortised bisimulation and provide a sound and complete proof system for amortised observational congruence. It turns out that it is sufficient to add the probabilistic extensions of Milner's three τ-laws [Mil89] to the proof system of strong amortised bisimulation. They can also be considered as the reminiscent of Segala's τ-laws for non-alternating models in [BS01], of which the only difference is that here they are decorated with indexed inequalities.

Related Works. Developing sound and complete proof systems for behavioral equivalences has long been a research focus in the process algebra community. Among the works on analyzing quantitative systems, a lot of attention has been devoted to the analysis of probabilistic behaviors. For the classical notion of probabilistic bisimulations, Bandini and Segala in [BS01] gave axiomatizations for strong and weak behavioral equivalences on simple probabilistic automata. Deng *et al.* provided similar axiomatizations for a language that includes parallel composition and (guarded) recursion on simple probabilistic automata [DPP05] and probabilistic automata [DP05], respectively. For more related work, we refer to references therein.

5.1 A simple probabilistic process algebra

Following [BS01] we consider a subset of probabilistic process algebra. Compared with the PPA introduced in Section 2.3, it excludes the recursion, the parallel composition and the restriction operators and requires the probabilistic choice operator to be prefixed.

Let I be a set of finite indices, a range over a finite set A of *labels*. Let $\tau \in A$, we recall that τ is the *silent action*. Let *NProc* denote the set of *nondeterministic processes*, ranged over by E, F or G, and *PProc* denote the set of *probabilistic processes*, ranged over by P. Finally, let $Proc \triangleq NProc \cup PProc$ denote the set of *processes*, ranged over by Q. The syntax of our probabilistic process algebra (PPA) is defined by the following BNF grammar:

$$\begin{aligned} E, F &::= \mathbf{0} \mid E + F \mid a.P \\ P &::= \bigoplus_{i \in I} p_i E_i \end{aligned}$$

Process $a.P$ performs action a and then becomes P which must be a *probabilistic choice*.

The semantics of a SPPA term is a probabilistic automaton defined according to the rules in Fig. 5.1, essentially following the non-alternating model defined in [BS01]. $P \mapsto \mu$ simply states that the probability distribution associated with P is μ.

We define the *depth* of a process Q, $d(Q)$, to be the maximum number of nested probabilistic choice operators in Q. Formally,

$$\begin{aligned} d(\mathbf{0}) &= 0 \\ d(\textstyle\bigoplus_{j \in J} p_j E_j) &= 1 + \max_{j \in J} d(E_j) \\ d(a.P) &= d(P) \\ d(\textstyle\sum_{i \in I} E_i) &= \max_{i \in I} d(E_i) \end{aligned}$$

5.2 Amortised probabilistic bisimulation

We denote by $Q \prec^{(\epsilon,c)} Q'$, where $|c| \leq \epsilon$, if there exists an ϵ-amortised bisimulation $\mathcal{R}$ such that $(Q, Q', c) \in \mathcal{R}$. We refer to Def. 4.3.3 in Section 4.2 for the definition of amortised bisimulation.

5.2.1 Basic properties

Some basic properties of $\prec^{(\epsilon,c)}$ are given in the next proposition.

$$\text{lchoice}\quad \frac{E \xrightarrow{a} \mu}{E + F \xrightarrow{a} \mu} \qquad \text{rchoice}\quad \frac{F \xrightarrow{a} \mu}{E + F \xrightarrow{a} \mu}$$

$$\text{idle}\quad \frac{-}{\bigoplus_{i\in I} p_i E_i \mapsto \sum_{i\in I} p_i E_i} \qquad \text{p-idle}\quad \frac{P \mapsto \mu}{P \xrightarrow{\tau} \mu}$$

$$\text{prefix}\quad \frac{P \mapsto \mu}{a.P \xrightarrow{a} \mu}$$

Figure 5.1: The operational semantics of SPPA.

Proposition 5.2.1. *The following hold:*

1. $Q \prec^{(0,0)} Q$;
2. $Q \prec^{(\epsilon,c)} Q'$ *iff* $Q' \prec^{(\epsilon,-c)} Q$;
3. *If* $Q_1 \prec^{(\epsilon_1,c_1)} Q_2 \prec^{(\epsilon_2,c_2)} Q_3$ *then* $Q_1 \prec^{(\epsilon_1+\epsilon_2,c_1+c_2)} Q_3$;
4. $\prec^{(\epsilon_1,c_1)} \subseteq \prec^{(\epsilon_2,c_2)}$ *where* $\epsilon_1 \leq \epsilon_2$ *and* $|c_2 - c_1| \leq \epsilon_2 - \epsilon_1$;
5. *For any* ϵ *and any* $|c| \leq \epsilon$, $\prec^{(0,0)} \subseteq \prec^{(\epsilon,c)}$.

Proof sketch. The proof proceeds by constructing an amortised bisimulation relation $\mathcal{R}'$ witnessing each pair of related processes. The proofs for Cases *1-3* are analogous to the proofs in Prop. 4.3.6. We discuss the remaining two cases:

4 Assuming $Q \prec^{(\epsilon_1,c_1)} Q'$, there exist an ϵ_1-amortised bisimulation $\mathcal{R}$ and $|c_1| \leq \epsilon_1$ such that $(Q, Q', c_1) \in \mathcal{R}$. Let $\mathcal{R}' \subseteq \mathit{Proc} \times \mathit{Proc} \times [-\epsilon_2, \epsilon_2]$ where $\epsilon_1 \leq \epsilon_2$:

$$\{\ (Q, Q', c_2) \mid \exists c_1, (Q, Q', c_1) \in \mathcal{R}\ \wedge\ |c_2 - c_1| \leq \epsilon_2 - \epsilon_1\ \}.$$

It is routine to verify that $\mathcal{R}'$ is the required amortised bisimulations.

$$\textbf{Weak1}\ \frac{E \xrightarrow{a} \mu}{E \stackrel{a}{\Longrightarrow} \mu}\ a \in A \qquad \textbf{Weak2}\ \frac{E \xrightarrow{\tau} \mu}{E \Longrightarrow \mu} \qquad \textbf{Weak3}\ \frac{-}{E \Longrightarrow \delta(E)}$$

$$\textbf{Weak4}\ \frac{E \stackrel{a}{\Longrightarrow} \mu \qquad \forall E_i \in supp(\mu).\ E_i \Longrightarrow \mu_i}{E \stackrel{a}{\Longrightarrow} \sum_{E_i \in supp(\mu)} \mu(E_i)\mu_i}\ a \in A \cup \{\varepsilon\}$$

$$\textbf{Weak5}\ \frac{E \Longrightarrow \mu \qquad \forall E_i \in supp(\mu).\ E_i \stackrel{a}{\Longrightarrow} \mu_i}{E \stackrel{a}{\Longrightarrow} \sum_{E_i \in supp(\mu)} \mu(E_i)\mu_i}\ a \in A \cup \{\varepsilon\}$$

Figure 5.2: Weak transitions

5 An immediate consequence of *4*. □

The following proposition lists the congruence properties of $\prec^{(\epsilon,c)}$, namely $\prec^{(\epsilon,c)}$ is substitutive under all SPPA combinators.

Proposition 5.2.2 (Substitutivity)**.** *Let* $E \prec^{(\epsilon,c)} E'$ *and* $P \prec^{(\epsilon,c)} P'$*. Then*

1. $E + F \prec^{(\epsilon,c)} E' + F$
2. $\bigoplus_{i \in 1..n-1} p_i E_i \oplus p_n E \prec^{(\epsilon,c)} \bigoplus_{i \in 1..n-1} p_i E_i \oplus p_n E'$
3. $a.P \prec^{(\epsilon,c)} a.P'$

Proof sketch. Analogous to the proofs of Cases *1-3* in Prop. 4.5.1. □

5.3 Weak amortised probabilistic bisimulation

In this section we shall introduce the notion of weak amortised bisimulations and amortised observational congruence.

The definition of *weak transitions* [BS01] is given in Fig. 5.2. Rules **Weak1** − **3** say that a weak transition either starts from a strong transition,

or is a self-loop of a process, while **Weak4** and **Weak5** allow internal transitions after and before an a-action to be absorbed. For any $F \in NProc$, $\big(\sum_{E_i \in supp(\mu)} \mu(E_i)\mu_i\big)(F) = \sum_{E_i \in supp(\mu)} \mu(E_i)\mu_i(F)$.

Note that $E \Longrightarrow \mu$ means that there are zero or more τ actions performed, while $E \stackrel{\tau}{\Longrightarrow} \mu$ means at least one τ action is performed; In **Weak4** and **Weak5**, ε represents the empty string, $\stackrel{\varepsilon}{\Longrightarrow}$ is actually $\Longrightarrow$ where ε is omitted for simplicity.

Following [Mil89], given a sequence $t \in A^*$ of labels, we denote by $\widehat{t}$ the sequence obtained from t by removing all occurrences of τ in t.

Definition 5.3.1 (Weak amortised bisimulation)**.** *Given $\epsilon \geq 0$ and $|c| \leq \epsilon$. A relation $\mathcal{R} \subseteq Proc \times Proc \times [-\epsilon, \epsilon]$ is an ϵ-*weak amortised bisimulation *if for all $(Q, Q', c) \in \mathcal{R}$:*

1. *$Q \xrightarrow{a} \mu$ implies $Q' \stackrel{\widehat{a}}{\Longrightarrow} \mu'$ and $\mu \; \mathcal{L}^A(\mathcal{R}, c) \; \mu'$;*
2. *$Q' \xrightarrow{a} \mu'$ implies $Q \stackrel{\widehat{a}}{\Longrightarrow} \mu$ and $\mu \; \mathcal{L}^A(\mathcal{R}, c) \; \mu'$.*

We write $Q \preccurlyeq^{(\epsilon,c)} Q'$, if there exists an ϵ-weak amortised bisimulation $\mathcal{R}$ such that $(Q, Q', c) \in \mathcal{R}$.

Note that although invisible actions are abstracted away in weak bisimulation, the contribution of the probabilities of each internal action is still taken into account when constituting a weak transition, as can be seen from Rules **Weak4-5** in Fig. 5.2. The performance of every invisible action (if is not an empty move) will alter the probability distribution of the resulting weak transition, thus affecting the variation budget during mutual simulations.

Let

$$\preccurlyeq^{\epsilon} = \bigcup \{\mathcal{R} : \; \mathcal{R} \text{ is an } \epsilon\text{-weak amortised bisimulation}\}$$

We can check by Def. 5.3.1 that $\preccurlyeq^{\epsilon}$ is itself an ϵ-weak amortised bisimulation, and is the largest one.

5.3.1 Basic properties of $\preccurlyeq$

This subsection aims at showing the metrical properties of $\preccurlyeq^{\epsilon}$, namely reflexivity, symmetry and triangle inequality.

The following lemma says an interesting fact that two processes that satisfy both (ϵ, c_1) and (ϵ, c_3) indexed weak amortised bisimulation satisfy (ϵ, c_2) indexed weak amortised bisimulation for any $c_3 < c_2 < c_1$.

Lemma 5.3.2. *If $Q \preccurlyeq^{(\epsilon,c_1)} Q'$, $Q \preccurlyeq^{(\epsilon,c_3)} Q'$ and there exists c_2 satisfying $c_3 < c_2 < c_1$, then $Q \preccurlyeq^{(\epsilon,c_2)} Q'$.*

Proof. Let $\mathcal{R} \subseteq Proc \times Proc \times [-\epsilon, \epsilon]$ be:

$$\{\ (Q, Q', c_2) |\ \exists c_1, c_3, c_3 < c_2 < c_1 \wedge (Q, Q', c_1) \in \preccurlyeq^{\epsilon} \wedge (Q, Q', c_3) \in \preccurlyeq^{\epsilon}\ \}.$$

Assume $(Q, Q', c_2) \in \mathcal{R}$, if $Q \stackrel{a}{\Longrightarrow} \mu$, by $(Q, Q', c_1) \in \preccurlyeq^{\epsilon}$ there exists $Q' \stackrel{a}{\Longrightarrow} \mu'$ and $\mu \mathcal{L}^{\mathcal{A}}(\preccurlyeq^{\epsilon}, c) \mu'$. Namely there are two weight functions ω_1 and ω_1' s.t. $\sum_F \omega_1(E, F) = \mu(E)$, $\sum_E \omega_1'(E, F) = \mu'(F)$ and

$$(E, F, c_1 + \ln \frac{\omega_1(E, F)}{\omega_1'(E, F)}) \in \preccurlyeq^{\epsilon}.$$

Analogously, by $(Q, Q', c_3) \in \preccurlyeq^{\epsilon}$, there are another two weight functions ω_3 and ω_3' s.t. $\sum_F \omega_3(E, F) = \mu(E)$, $\sum_E \omega_3'(E, F) = \mu'(F)$ and

$$(E, F, c_3 + \ln \frac{\omega_3(E, F)}{\omega_3'(E, F)}) \in \preccurlyeq^{\epsilon}.$$

We define weight functions ω_2 and ω_2' s.t. $\omega_2(E, F) = \frac{1}{2}\omega_1(E, F) + \frac{1}{2}\omega_3(E, F)$ and $\omega_2'(E, F) = \frac{1}{2}\omega_1'(E, F) + \frac{1}{2}\omega_3'(E, F)$. By the definition of $\mathcal{R}$, it holds

$$(E, F, c_2 + \ln \frac{\omega_2(E, F)}{\omega_2'(E, F)}) \in \mathcal{R}.$$

Hence, $\mathcal{R}$ is an ϵ-weak amortised bisimulation, namely $Q \preccurlyeq^{(\epsilon,c_2)} Q'$ as required. □

In order to prove the triangle inequality in the setting of weak transitions, we need to ensure that weak transitions are transitive, as shown in the next lemma.

Lemma 5.3.3. *If $Q \preccurlyeq^{(\epsilon,c)} Q'$, then $Q \stackrel{a}{\Longrightarrow} \mu$ implies there exists μ': $Q' \stackrel{\widehat{a}}{\Longrightarrow} \mu'$ and $\mu\ \mathcal{L}^A(\preccurlyeq^\epsilon, c)\ \mu'$.*

Proof. The proof proceeds by induction on the length of the strong transitions which $\stackrel{a}{\Longrightarrow}$ is composed of.

- The basis is when the length equals to 1, $Q \stackrel{a}{\Longrightarrow} \mu$ degenerates to a strong transition and the result is trivial.

- For the inductive step: consider the case where the length equals to N and the first move is $\stackrel{a}{\longrightarrow}$, i.e. $\stackrel{a}{\Longrightarrow} = \stackrel{a}{\longrightarrow}\stackrel{\tau}{\Longrightarrow}$. Note that the other case is when the first move is $\stackrel{\tau}{\longrightarrow}$, the essence of its proof is analogous and omitted for simplicity.

  Assume $Q \stackrel{a}{\longrightarrow} \nu$ and $\forall E_i \in supp(\nu), E_i \stackrel{\tau}{\Longrightarrow} \mu_i$ whose length of strong transitions is less than N, we have $\mu = \sum_{E_i \in supp(\nu)} \nu(E_i)\mu_i$. There exists a weak transition $Q' \stackrel{\widehat{a}}{\Longrightarrow} \nu'$ and $\nu\ \mathcal{L}^A(\preccurlyeq^\epsilon, c)\ \nu'$. Namely, there are two weights functions ω, ω' such that for each $E_i \in supp(\nu)$, $\sum_{F_j} \omega(E_i, F_j) = \nu(E_i)$; for each $F_j \in supp(\nu')$, $\sum_{E_i} \omega'(E_i, F_j) = \nu'(F_j)$; and $\omega(E_i, F_j) = 0$ iff $\omega'(E_i, F_j) = 0$; if $\omega(E_i, F_j) > 0$, then

$$(E_i, F_j, c + \ln \frac{\omega(E_i, F_j)}{\omega'(E_i, F_j)}) \in \preccurlyeq^\epsilon .$$

  By $E_i \stackrel{\tau}{\Longrightarrow} \mu_i$ and inductive hypothesis, there exists a weak transition $F_j \stackrel{\tau}{\Longrightarrow} \mu'_j$ such that $\mu_i\ \mathcal{L}^A(\preccurlyeq^\epsilon, c + \ln \frac{\omega(E_i,F_j)}{\omega'(E_i,F_j)})\ \mu'_j$. Namely, there exist weight functions $\gamma_{ij}, \gamma'_{ij}$ s.t. $\forall G, G', \sum_{G'} \gamma_{ij}(G, G') = \mu_i(G)$, $\sum_G \gamma'_{ij}(G, G') = \mu'_j(G')$; and $\gamma_{ij}(G, G') = 0$ iff $\gamma'_{ij}(G, G') = 0$; if $\gamma_{ij}(G, G') > 0$, then

$$\big(G, G', c + \ln \frac{\omega(E_i, F_j)}{\omega'(E_i, F_j)} + \ln \gamma_{ij}(G, G') - \ln \gamma'_{ij}(G, G')\ \big) \in \preccurlyeq^\epsilon,$$

  which equals to

$$\big(\ G, G', c + \ln \frac{\omega(E_i, F_j) \cdot \gamma_{ij}(G, G')}{\omega'(E_i, F_j) \cdot \gamma'_{ij}(G, G')}\ \big) \in \preccurlyeq^\epsilon . \tag{5.1}$$

  Let $\mu' = \sum_{F_j \in supp(\nu')} \nu'(F_j)\mu'_j$. We shall construct desirable weight functions π and π' for μ and μ' out of ω, ω', γ_{ij} and γ'_{ij} in such a way that

$\mu\ \mathcal{L}^A(\preccurlyeq^\epsilon, c)\ \mu'$. Let

$$\pi(G, G') = \sum_{E_i, F_j} \omega(E_i, F_j)\gamma_{ij}(G, G')$$
$$\pi'(G, G') = \sum_{E_i, F_j} \omega'(E_i, F_j)\gamma'_{ij}(G, G').$$

Then we have

$$\begin{aligned}
&\textstyle\sum_{G'} \pi(G, G') \\
&= \textstyle\sum_{G', E_i, F_j} \omega(E_i, F_j)\gamma_{ij}(G, G') && \text{Definition of } \pi \\
&= \textstyle\sum_{E_i, F_j} \omega(E_i, F_j) \sum_{G'} \gamma_{ij}(G, G') && \text{Commutativity} \\
&= \textstyle\sum_{E_i, F_j} \omega(E_i, F_j)\mu_i(G) && \text{Definition of } \gamma_{ij} \\
&= \textstyle\sum_{E_i} \mu_i(G)\nu(E_i) && \text{Definition of } \omega \\
&= \mu(G) && \text{Definition of } \mu
\end{aligned}$$

Analogously, we can check that $\sum_G \pi'(G, G') = \mu'(G)$; and $\pi(G, G') = 0$ iff $\pi'(G, G') = 0$; and if $\pi(G, G') > 0$, by Equation (5.1), Lemma 5.3.2 and

$$\min_{E_i, F_j} \ln \frac{\omega(E_i, F_j)\gamma_{ij}(G, G')}{\omega'(E_i, F_j)\gamma'_{ij}(G, G')} \leq \ln \frac{\pi(G, G')}{\pi'(G, G')} \leq \max_{E_i, F_j} \ln \frac{\omega(E_i, F_j)\gamma_{ij}(G, G')}{\omega'(E_i, F_j)\gamma'_{ij}(G, G')},$$

it follows:

$$\Big(G, G', c + \ln \frac{\pi(G, G')}{\pi'(G, G')}\Big) \in \preccurlyeq^\epsilon .$$

Thus π, π' are the two required weight functions and $\mu\ \mathcal{L}^{\mathcal{A}}(\preccurlyeq^\epsilon, c)\ \mu'$ holds.

□

After proving that weak transitions are transitive, we now can show the following basic properties of $\preccurlyeq$:

Proposition 5.3.4. *The following hold:*

1. $Q \preccurlyeq^{(0,0)} Q$;
2. $Q \preccurlyeq^{(\epsilon,c)} Q'$ *iff* $Q' \preccurlyeq^{(\epsilon,-c)} Q$;
3. *If* $Q_1 \preccurlyeq^{(\epsilon_1,c_1)} Q_2 \preccurlyeq^{(\epsilon_2,c_2)} Q_3$ *then* $Q_1 \preccurlyeq^{(\epsilon_1+\epsilon_2,c_1+c_2)} Q_3$.

Proof sketch. The proof of cases *1*, *2* and *3* are similar to the proofs of *1*, *2* and *3* of Proposition 5.2.1, respectively, where case *3* needs the help of Lemma 5.3.3. □

The following proposition shows that the first τ-transition can be ignored for weak bisimulation:

Proposition 5.3.5. $\tau.\Delta(E) \preccurlyeq^{(0,0)} E$.

Proof sketch. The essence of the proof is the observation that

$$\mathcal{R} = \{(\tau.\Delta(E), E, 0)\} \cup Id_{Proc}$$

is a 0-weak amortised bisimulation. □

5.3.2 Amortised observational congruence

It is well known that weak bisimulation is not preserved by the external choice operator $+$ [Mil89]. A typical example is

$$\tau.\Delta(b.\Delta(\mathbf{0})) \preccurlyeq^{(0,0)} b.\Delta(\mathbf{0}),$$

while

$$a.\Delta(\mathbf{0}) + \tau.\Delta(b.\Delta(\mathbf{0})) \not\preccurlyeq^{(0,0)} a.\Delta(\mathbf{0}) + b.\Delta(\mathbf{0}).$$

Hence, as usual, we define observational congruence on top of weak bisimulation $\preccurlyeq^{\epsilon}$ as follows.

Definition 5.3.6 (Amortised observational congruence)**.** *Given $\epsilon \geq 0$ and $|c| \leq \epsilon$. Q and Q' are (ϵ, c)-*amortised observationally congruent*, written $Q \preceq^{(\epsilon,c)} Q'$, if for all $a \in A$*

1. *$Q \xrightarrow{a} \mu$ implies $Q' \stackrel{a}{\Longrightarrow} \mu'$ and $\mu \; \mathcal{L}^A(\preccurlyeq^{\epsilon}, c) \; \mu'$;*
2. *$Q' \xrightarrow{a} \mu'$ implies $Q \stackrel{a}{\Longrightarrow} \mu$ and $\mu \; \mathcal{L}^A(\preccurlyeq^{\epsilon}, c) \; \mu'$.*

Def. 5.3.6 differs from Def. 5.3.1 only in one aspect: $Q' \stackrel{a}{\Longrightarrow} \mu'$, instead of $Q' \stackrel{\widehat{a}}{\Longrightarrow} \mu'$, is required to match $Q \xrightarrow{a} \mu$. This implies that a τ-transition from a process must be matched by *at least one* τ-transition from the other.

Note also that this is required only for the first transitions. For their corresponding derivative distributions μ and μ', they are required to satisfy the lifting operation just w.r.t $\preccurlyeq$, rather than $\preceq$.

The following two propositions establish the relations between $\preccurlyeq$ and $\preceq$. Proposition 5.3.8 will play an important role when proving the completeness of the proof system for $\preceq$ in Section 5.5.

Proposition 5.3.7. *$Q \preceq^{(\epsilon,c)} Q'$ implies $Q \preccurlyeq^{(\epsilon,c)} Q'$.*

Proof. Straightforward by their definitions. □

Proposition 5.3.8. *$Q \preccurlyeq^{(\epsilon,c)} Q'$ iff $Q \preceq^{(\epsilon,c)} Q'$ or $Q \preceq^{(\epsilon,c)} \tau.\Delta(Q')$ or $\tau.\Delta(Q) \preceq^{(\epsilon,c)} Q'$.*

Proof. ($\Longleftarrow$) By Proposition 5.3.7, we have $Q \preccurlyeq^{(\epsilon,c)} Q'$ or $Q \preccurlyeq^{(\epsilon,c)} \tau.\Delta(Q')$ or $\tau.\Delta(Q) \preccurlyeq^{(\epsilon,c)} Q'$. Together with Proposition 5.3.4 (*2*), (*3*) and Proposition 5.3.5, we obtain $Q \preccurlyeq^{(\epsilon,c)} Q'$.

($\Longrightarrow$) Assume $Q \preccurlyeq^{(\epsilon,c)} Q'$, we need to consider three cases. First, suppose that $Q \xrightarrow{\tau} \mu$ for some μ, Q' has no τ-transition but to match it with a null label $Q' \stackrel{\varepsilon}{\Longrightarrow} \delta(Q')$ and $\mu \; \mathcal{L}^A(\preccurlyeq^\epsilon, c) \; \delta(Q')$, then it is easy to see that $Q \preceq^{(\epsilon,c)} \tau.\Delta(Q')$. Second, suppose that $Q' \xrightarrow{\tau} \mu'$ for some μ', Q has no τ-transition but to match it with a null label $Q \stackrel{\varepsilon}{\Longrightarrow} \delta(Q)$ and $\delta(Q) \; \mathcal{L}^A(\preccurlyeq^\epsilon, c) \; \mu'$, then similarly we show that $\tau.\Delta(Q) \preccurlyeq^{(\epsilon,c)} Q'$. Thirdly, neither of these conditions holds, namely, if Q performs a τ-transition, then Q' also has a non-zero τ-transition to match it, and vise versa. Since $\preceq$ and $\preccurlyeq$ naturally coincide on performing visible transitions, we show that $Q \preceq^{(\epsilon,c)} Q'$ as required. □

Similar to the strong case, we have

Proposition 5.3.9. *The following hold:*

1. $Q \preceq^{(0,0)} Q$;
2. $Q \preceq^{(\epsilon,c)} Q'$ *iff* $Q' \preceq^{(\epsilon,-c)} Q$;
3. *If* $Q_1 \preceq^{(\epsilon_1,c_1)} Q_2 \preceq^{(\epsilon_2,c_2)} Q_3$ *then* $Q_1 \preceq^{(\epsilon_1+\epsilon_2,c_1+c_2)} Q_3$;
4. $\preceq^{(\epsilon_1,c_1)} \subseteq \preceq^{(\epsilon_2,c_2)}$ *where* $\epsilon_1 \leq \epsilon_2$ *and* $|c_2 - c_1| \leq \epsilon_2 - \epsilon_1$.

Proof sketch. The proofs of the first three cases proceed analogously as in the proof of Proposition 5.3.4. Case *4*'s proof is similar to the proof of Proposition 5.2.1. □

Proposition 5.3.10 (Substitutivity)**.** *Let* $E \preceq^{(\epsilon,c)} E'$ *and* $P \preceq^{(\epsilon,c)} P'$. *Then*

1. $E + F \preceq^{(\epsilon,c)} E' + F$;
2. $\bigoplus_{i\in 1..n-1} p_i E_i \oplus p_n E \preceq^{(\epsilon,c)} \bigoplus_{i\in 1..n-1} p_i E_i \oplus p_n E'$;
3. $a.P \preceq^{(\epsilon,c)} a.P'$.

Proof sketch. The proofs for Clause *1* and *3* are straightforward. We detail below the proof of Clause *2*.

By assumption, there exists an ϵ-amortised observational congruence $\mathcal{R}$ s.t. $(E, E', c) \in \mathcal{R}$. Let

$$\mathcal{R}' = \{ \; (\bigoplus_{i\in 1..n-1} p_i E_i \oplus p_n E, \bigoplus_{i\in 1..n-1} p_i E_i \oplus p_n E', c) \; \} \cup \mathcal{R} \cup Id_{Proc}.$$

Let $\bigoplus_{i\in 1..n-1} p_i E_i \oplus p_n E \stackrel{a}{\Longrightarrow} \mu$, we have to show that there exists also a weak transition $\bigoplus_{i\in 1..n-1} p_i E_i \oplus p_n E' \stackrel{a}{\Longrightarrow} \mu'$ such that $\mu \; \mathcal{L}^A(\preccurlyeq^\epsilon, c) \; \mu'$.

Assume that $\mu = \sum_{i\in 1..n-1} p_i \mu_i \oplus p_n \nu$ where for $i \in 1..n-1$, $E_i \stackrel{a}{\Longrightarrow} \mu_i$ and $E \stackrel{a}{\Longrightarrow} \nu$. Since $(E, E', c) \in \mathcal{R}$, there exists $E' \stackrel{a}{\Longrightarrow} \nu'$ such that $\nu \; \mathcal{L}^A(\preccurlyeq^\epsilon, c) \; \nu'$. Namely, there exist two weight functions $\omega, \omega' : NProc \times NProc \to [0,1]$ such that for each $G \in NProc$, $\sum_{G'} \omega(G, G') = \nu(G)$; for each $G' \in NProc$, $\sum_{G} \omega'(G, G') = \nu'(G')$; and $\omega(G, G') = 0$ iff $\omega'(G, G') = 0$; if $\omega(G, G') > 0$ then

$$(G, G', c + \ln \frac{\omega(G, G')}{\omega'(G, G')}) \in \preccurlyeq^\epsilon .$$

Let $\mu' = \sum_{i\in 1..n-1} p_i \mu_i \oplus p_n \nu'$. We shall construct two desirable weight functions π, π' for μ and μ' out of ω, ω'.

We rename the states in the support set of ν and ν', such that for any $i \in 1..n-1$, $supp(\mu_i) \cap supp(\nu) = \emptyset$ and also $supp(\mu_i) \cap supp(\nu') = \emptyset$.

Let π, π':

$$\pi(G, G') = p_n \cdot \omega(G, G') \quad \text{if } G \in supp(\nu) \text{ and } G' \in supp(\nu');$$

$$\pi'(G, G') = p_n \cdot \omega'(G, G') \quad \text{if } G \in supp(\nu) \text{ and } G' \in supp(\nu');$$

$$\pi(G, G') = \pi'(G, G') = \begin{cases} \sum_{i \in 1..n-1} p_i \cdot \mu_i(G) & \text{if } G = G' \notin supp(\nu) \cup supp(\nu'); \\ 0 & \text{otherwise.} \end{cases}$$

We can check by definition that π, π' satisfy the lifting condition, and $\mu \ \mathcal{L}^A(\preccurlyeq^\epsilon, c) \ \mu'$ holds as required. □

5.4 Proof system $\mathcal{A}_1$ for amortised bisimulation

This section is devoted to presenting a proof system $\mathcal{A}_1$ for strong amortised probabilistic bisimulation $\prec$ and proving its soundness and completeness. Soundness ensures the correctness of the proof system. Completeness asserts that all amortised bisimulations can be syntactically derived from the proof system.

The statements of the proof system are of the form $(\epsilon, c) \rhd Q \prec Q'$. The proof system consists of axioms and inference rules, shown in Fig. 5.3 and Fig. 5.4, in the spirit of [HL93].

Axioms A′1-4 are reminiscent of the monoid laws for bisimulation. Axiom A′5 will be used to transform probabilistic processes terms into normal forms. The two axioms A′6-7 allow us to permute and merge branches of a probabilistic choice.

The usual rules for equality (substitutivity, reflexivity, symmetry and transitivity) hold only for $(0, 0)$-indexed pairs of processes, and can not straightforwardly apply when reasoning on non-zero indexed inequalities. Hence we need **Subs**, **Refl**, **Symm** and a more general rule **Triangle** (short for triangle inequality) to regulate indexing inequalities when rewriting processes.

The **Weakening** rule does not deal with any specific operator in the language. It is a kind of structural rule used to glue proofs together. Rules

A1	$E + \mathbf{0} = E$
A2	$E + E = E$
A3	$E + F = F + E$
A4	$(E + F) + G = E + (F + G)$
A5	$\tau.P = P$
A6	$p_1E_1 \oplus \cdots \oplus p_iE_i \oplus p_jE_j \oplus \cdots \oplus p_nE_n$
	$= p_1E_1 \oplus \cdots \oplus p_jE_j \oplus p_iE_i \oplus \cdots \oplus p_nE_n$
A7	$\bigoplus_{i\in I} p_iE_i \oplus pE \oplus qE = \bigoplus_{i\in I} p_iE_i \oplus (p+q)E$

Figure 5.3: The proof system $\mathcal{A}_1$: Axioms

Sum, **Prefix** and **Prob** manipulates nondeterministic choice, prefixing and probabilistic choice, respectively. The side condition of **Prob** requires that probabilistic combinations be permitted only when for every c_i indexing a related pair $E_i \prec F_i$ and the corresponding probability weights p_i and q_i, the updated index $c_i - \ln p_i + \ln q_i$ agrees on the same value c. The side condition ensures that the resulting pair of probabilistic distributions satisfies the conditions of the lifting operation (Def. 4.3.2).

Theorem 5.4.1 (Soundness of $\mathcal{A}_1$)**.** *If* $\mathcal{A}_1 \vdash (\epsilon, c) \rhd Q \prec Q'$ *then* $Q \prec^{(\epsilon,c)} Q'$.

Proof. The proof proceeds by induction on the length of the derivation for $(\epsilon, c) \rhd Q \prec Q'$, with a case analysis on the last applied axiom or inference rule:

When the length equals to 1, for axioms (A′1-7), let $\mathcal{R} = \{(Q, Q', 0)\} \cup Id_{Proc}$. It is easy to check that $\mathcal{R}$ is a 0-amortised bisimulation. **Refl** is trivial from Proposition 5.2.1 (*1*).

For the inductive step:

- The soundness of the rules **Subs**, **Symm**, **Triangle** and **Weakening** are supported by Proposition 5.2.2, Proposition 5.2.1 (*2*), (*3*) and (*4*),

A′1-7	$(0,0) \rhd Q \prec Q'$, where $Q = Q'$ is an instance of one of the axioms A1-A7
Subs	If $(0,0) \rhd Q \prec Q'$, then Q and Q' are substitutive under all PPA operators.

Refl $(0,0) \rhd Q \prec Q$ **Symm** $\dfrac{(0,0) \rhd Q \prec Q'}{(0,0) \rhd Q' \prec Q}$

Triangle $\dfrac{(\epsilon_1, c_1) \rhd Q_1 \prec Q_2 \qquad (\epsilon_2, c_2) \rhd Q_2 \prec Q_3}{(\epsilon_1 + \epsilon_2, c_1 + c_2) \rhd Q_1 \prec Q_3}$

Weakening $\dfrac{(\epsilon_1, c_1) \rhd Q \prec Q'}{(\epsilon_2, c_2) \rhd Q \prec Q'}$ $\epsilon_1 \leq \epsilon_2$ and $|c_2 - c_1| \leq \epsilon_2 - \epsilon_1$

Sum $\dfrac{(\epsilon, c) \rhd E_1 \prec F_1 \qquad (\epsilon, c) \rhd E_2 \prec F_2}{(\epsilon, c) \rhd E_1 + E_2 \prec F_1 + F_2}$

Prefix $\dfrac{(\epsilon, c) \rhd P_1 \prec P_2}{(\epsilon, c) \rhd a.P_1 \prec a.P_2}$

Prob $\dfrac{\forall i \in I, \quad (\epsilon, c_i) \rhd E_i \prec F_i}{(\epsilon, c) \rhd \bigoplus_{i \in I} p_i E_i \prec \bigoplus_{i \in I} q_i F_i}$ $\forall i \in I, c_i - \ln p_i + \ln q_i = c, |c| \leq \epsilon$

Figure 5.4: The proof system $\mathcal{A}_1$: Inference Rules

respectively.

- **Sum** By the induction hypothesis, there exist ϵ-amortised bisimulations $\mathcal{R}_1$ and $\mathcal{R}_2$ s.t. $(E_1, F_1, c) \in \mathcal{R}_1$ and $(E_2, F_2, c) \in \mathcal{R}_2$. Let

$$\mathcal{R} = \{(E_1 + E_2, F_1 + F_2, c)\} \cup \mathcal{R}_1 \cup \mathcal{R}_2.$$

- **Prefix** By the induction hypothesis, there exists an ϵ-amortised bisimulation $\mathcal{R}'$ s.t. $(P_1,\ P_2, c) \in \mathcal{R}'$. Let

$$\mathcal{R} = \{(a.P_1, a.P_2, c)\} \cup \mathcal{R}'.$$

- **Prob** By the induction hypothesis, there exists an ϵ-amortised bisimulation $\mathcal{R}_i$ and $|c_i| \leq \epsilon$ s.t. $(E_i, F_i, c_i) \in \mathcal{R}_i$. Let

$$\mathcal{R} = \{(\bigoplus_{i\in I} p_i E_i, \bigoplus_{i\in I} q_i F_i, c)\} \cup \bigcup_{i\in I} \mathcal{R}_i.$$

  For any $i, j \in I$, let $\omega(E_i, F_j) = p_i$ if $i = j$; otherwise 0; let $\omega'(E_i, F_j) = q_i$ if $i = j$, otherwise 0. By the definition of lifting operation, ω and ω' can be shown to be two desirable weight functions between the two probabilistic processes.

  It is routine to check that $\mathcal{R}$'s defined above are all ϵ-amortised bisimulations.

□

The proof of the completeness result is similar to the corresponding proof for CCS [Mil89]: Processes are transformed to equivalent normal forms. Equivalence here means two processes are $(0, 0)$-indexed. Then processes are compared almost syntactically piece by piece, and finally duplicate terms are merged. We first give the notion of *normal form* that will be needed in the proof:

Definition 5.4.2. *A process Q is in* normal form*(NF) if*

- *either $Q \equiv \mathbf{0}$;*
- *or $Q \equiv \sum_{i\in I} a_i.\bigoplus_{j\in J_i} p^i_j E^i_j$, where each E^i_j is also in normal form.*

Lemma 5.4.3. *For any Q, there is a normal form $\widehat{Q}$ such that*

$$\mathcal{A}_1 \vdash (0, 0) \rhd Q \prec \widehat{Q}.$$

We will use $\widehat{Q}$ to denote one normal form of Q.

Proof. The proof proceeds by induction on the depth $d(Q)$ of Q.

If $d(Q) = 0$, then $Q \equiv \mathbf{0}$.

For the inductive step, two cases need to be considered:

- If $Q \in \mathit{NProc}$, assume $Q \equiv \sum_{i\in I} a_i.\bigoplus_{j\in J_i} p^i_j E^i_j$, by the induction hypothesis every E^i_j has a normal form. Using the proof system, the term $\mathbf{0}$ may be eliminated, and this results in a normal form.

- Otherwise $Q \in \mathit{PProc}$, assume $Q \equiv \bigoplus_{j\in J} p_j E_j$, by the induction hypothesis every E_i has a normal form $\widehat{E_i}$. We apply Axiom (A′5) and Rule **Symm** and obtain a normal form $\tau.Q'$ where $Q' \equiv \bigoplus_{j\in J} p_j \widehat{E_i}$ such that $\mathcal{A}_1 \vdash (0,0) \rhd Q \prec \tau.Q'$.

□

The following lemma says that normal forms inherit indices from their original processes, and vise versa.

Lemma 5.4.4. *$Q \prec^{(\epsilon,c)} Q'$ iff $\widehat{Q} \prec^{(\epsilon,c)} \widehat{Q'}$.*

Proof. By Lemma 5.4.3 and the soundness of the proof system, we have $Q \prec^{(0,0)} \widehat{Q}$ and $Q' \prec^{(0,0)} \widehat{Q'}$. By use of Proposition 5.2.1 (*2*) and (*3*), $Q \prec^{(\epsilon,c)} Q'$ iff $\widehat{Q} \prec^{(\epsilon,c)} \widehat{Q'}$. □

Theorem 5.4.5 (Completeness of $\mathcal{A}_1$)**.** *If $Q \prec^{(\epsilon,c)} Q'$ then $\mathcal{A}_1 \vdash (\epsilon, c) \rhd Q \prec Q'$.*

Proof. By Lemmas 5.4.3 and 5.4.4 we may assume Q and Q' are already in normal form. Let $Q \equiv \sum_{i\in I_1} a_i.\bigoplus_{j\in J_i} p^i_j E^i_j$ and $Q' \equiv \sum_{i\in I_2} a'_i.\bigoplus_{j\in J_i} q^i_j F^i_j$.

Assume that there exist an ϵ-amortised bisimulation $\mathcal{R}$ and $|c| \le \epsilon$ such that $(Q, Q', c) \in \mathcal{R}$. The proof proceeds by induction on the maximum depth of Q and Q'.

- If the maximum depth is 0 then Q and Q' are both $\mathbf{0}$, by **Refl** and **Weakening** it is easy to see that $\mathcal{A}_1 \vdash (\epsilon, c) \rhd \mathbf{0} \prec \mathbf{0}$.

- Otherwise the maximum depth is greater than 0, let $a.\bigoplus_{i\in 1..n} p_i E_i$ be a summand of Q. Then $Q \xrightarrow{a} \mu$ where $\mu = \biguplus_{i\in 1..n} p_i E_i$. Since $(Q, Q', c) \in \mathcal{R}$, there is some $\mu' = \biguplus_{j\in 1..m} q_j F_j$ such that $Q' \xrightarrow{a} \mu'$ and $\mu\ \mathcal{L}^A(\mathcal{R}, c)\ \mu'$. Namely, there exist two weight functions ω and ω' s.t. $\sum_{j\in 1..m} \omega(E_i, F_j) = p_i$, $\sum_{i\in 1..n} \omega'(E_i, F_j) = q_j$; for any i, j,

$\omega(E_i, F_j) = 0$ iff $\omega'(E_i, F_j) = 0$; if $\omega(E_i, F_j) > 0$ then $(E_i, F_j, c + \ln \frac{\omega(E_i,F_j)}{\omega'(E_i,F_j)}) \in \mathcal{R}$.

Then we have:

$$\mathcal{A}_1 \vdash (\epsilon, c + \ln \frac{\omega(E_i, F_j)}{\omega'(E_i, F_j)}) \rhd E_{ij} \prec F_{ij} \qquad \text{By the IH}$$

$$\mathcal{A}_1 \vdash (\epsilon, c) \rhd \bigoplus_{i,j} \omega(E_i, F_j) E_{ij} \prec \bigoplus_{i,j} \omega'(E_i, F_j) F_{ij} \qquad \textbf{Prob}$$

$$\mathcal{A}_1 \vdash (\epsilon, c) \rhd \bigoplus_{i \in 1..n} p_i E_i \prec \bigoplus_{j \in 1..m} q_j F_j \qquad \text{A}'6\text{-}7, \text{ Defs. of } \omega, \omega'$$

$$\mathcal{A}_1 \vdash (\epsilon, c) \rhd a.\bigoplus_{i \in 1..n} p_i E_i \prec a.\bigoplus_{j \in 1..m} q_j F_j \qquad \textbf{Prefix}$$

Similarly, every summand $a'.P'$ of Q' can be proved related to a summand $a'.P$ of Q with respect to (ϵ, c). Namely,

$$\mathcal{A}_1 \vdash (\epsilon, c) \rhd a'.P \prec a'.P'$$

By applying **Sum** to combine all these related pairs of summands, using (A′1-4), **Symm** and **Triangle** to reorder and regroup summands as necessary, eliminating duplicate summands, and finally it follows that $\mathcal{A}_1 \vdash (\epsilon, c) \rhd Q \prec Q'$.

□

5.5 Proof system $\mathcal{A}_2$ for amortised observational congruence

In this section we extend $\mathcal{A}_1$ to obtain a proof system for amortised observational congruence $\preceq$. It turns out that, as in the standard axiomatisations of bisimulations in CCS, it is sufficient to add the probabilistic versions of the τ-laws $\mathbf{T_1}, \mathbf{T_2}$ and $\mathbf{T_3}$, presented in Fig. 5.5, to $\mathcal{A}_1$. $\mathbf{T_2}$ and $\mathbf{T_3}$ can be considered as the reminiscent of $A6$ and $A7$ in the τ-laws for non-alternating models in [BS01], with the only difference that here they are decorated with indexed inequalities, rather than equalities.

$\mathbf{T_1}$	$(0,0) \rhd a.\big(\bigoplus_{i\in I} p_i E_i \oplus p\ \tau.\Delta(F)\big) \prec a.\big(\bigoplus_{i\in I} p_i E_i \oplus p\ F\big)$
$\mathbf{T_2}$	$(0,0) \rhd \tau.\bigoplus_{i\in I} p_i(E_i + a.P_i) + a.\bigoplus_{i\in I} p_i P_i \prec \tau.\bigoplus_{i\in I} p_i(E_i + a.P_i)$
$\mathbf{T_3}$	$(0,0) \rhd a.\big(\bigoplus_{i\in I} p_i(E_i + \tau.P_i) \oplus \bigoplus_{j\in J} p_j E_j\big) + a.\bigoplus_{i\in I\cup J} p_i P_i \prec$
	$a.\big(\bigoplus_{i\in I} p_i(E_i + \tau.P_i) \oplus \bigoplus_{j\in J} p_j E_j\big)$, where for $j \in J, P_j = \Delta(E_j)$

Figure 5.5: τ-laws

$$\text{Convex} \qquad \frac{\forall i \in I.\ P_i \mapsto \mu_i}{\bigoplus_{i\in I} p_i P_i \mapsto \sum_{i\in I} p_i \mu_i}$$

Figure 5.6: The operational semantics of the convex combinator

Note that a new combinator $\bigoplus_{i\in I} p_i P_i$ is used in Fig. 5.5, it is the *convex combination* of probabilistic processes [BS01]. Its semantics is given in Fig. 5.6. Observe that this combinator is not really new in the sense that it is, in fact, equivalent to $\bigoplus_{i\in I} p_i \tau.P_i$ in the weak semantics.

Let $\mathcal{A}_2 = \mathcal{A}_1 \cup \{\mathbf{T_1}, \mathbf{T_2}, \mathbf{T_3}\}$. The main body of this section is devoted to proving that $\mathcal{A}_2$ is sound and complete w.r.t. $\preceq$.

We first show that Rules **Sum**, **Prefix** and **Prob** are sound w.r.t. $\preceq$.

Proposition 5.5.1. *The following hold:*

1. *Let $i \in \{1,2\}$, $E_i \preceq^{(\epsilon,c)} F_i$, then $E_1 + E_2 \preceq^{(\epsilon,c)} F_1 + F_2$;*
2. *If $P_1 \preceq^{(\epsilon,c)} P_2$, then $a.P_1 \preceq^{(\epsilon,c)} a.P_2$;*
3. *Given $\forall i \in I$, $E_i \preceq^{(\epsilon,c_i)} F_i$, and $\bigoplus_{i\in I} p_i E_i$ and $\bigoplus_{i\in I} q_i F_i$ satisfy that for all i, $c_i - \ln p_i + \ln q_i = c$ and $|c| \leq \epsilon$, then $\bigoplus_{i\in I} p_i E_i \preceq^{(\epsilon,c)} \bigoplus_{i\in I} q_i F_i$.*

Proof. We sketch the proofs for Clauses *1* and *2*, while detail the proof of Clause *3*.

Clause *1* By assumption, there exist ϵ-amortised observational congruence $\mathcal{R}_1$ and $\mathcal{R}_2$ s.t. $(E_1, F_1, c) \in \mathcal{R}_1$ and $(E_2, F_2, c) \in \mathcal{R}_2$. Let

$$\mathcal{R} = \{(E_1 + E_2, F_1 + F_2, c)\} \cup \mathcal{R}_1 \cup \mathcal{R}_2.$$

Clause *2* By assumption, there exists an ϵ-amortised observational congruence $\mathcal{R}'$ s.t. $(P_1, P_2, c) \in \mathcal{R}'$. Let

$$\mathcal{R} = \{(a.P_1, a.P_2, c)\} \cup \mathcal{R}'.$$

It is routine to check that the two $\mathcal{R}$'s defined above are all ϵ-amortised observational congruence.

Clause *3* By assumption, for each c_i there exists an ϵ-amortised observational congruence $\mathcal{R}_i$ s.t. $(E_i, F_i, c_i) \in \mathcal{R}_i$. Let

$$\mathcal{R} = \{(\bigoplus_{i\in I} p_i E_i, \bigoplus_{i\in I} q_i F_i, c)\} \cup \bigcup_{i\in I} \mathcal{R}_i. \tag{5.2}$$

If $\bigoplus_{i\in I} p_i E_i \stackrel{a}{\Longrightarrow} \mu$, we have to show that there exists also a weak transition $\bigoplus_{i\in I} q_i F_i \stackrel{a}{\Longrightarrow} \mu'$ such that $\mu \; \mathcal{L}^A(\preccurlyeq^\epsilon, c) \; \mu'$.

Assume that $\mu = \sum\!\!\!\!\!\!\int_{i\in I} p_i \mu_i$ where for each i, $E_i \stackrel{a}{\Longrightarrow} \mu_i$. Since $(E_i, F_i, c_i) \in \mathcal{R}_i$, there exists $F_i \stackrel{a}{\Longrightarrow} \mu_i'$ such that $\mu_i \; \mathcal{L}^A(\preccurlyeq^\epsilon, c_i) \; \mu_i'$. Namely, there exist weight functions γ_i, γ_i' s.t. $\forall G, G', \sum_{G'} \gamma_i(G, G') = \mu_i(G)$, $\sum_G \gamma_i'(G, G') = \mu_i'(G')$

$$(G, G', c_i + \ln \gamma_i(G, G') - \ln \gamma_i'(G, G')) \in \preccurlyeq^\epsilon .$$

By assumption that $c_i - \ln p_i + \ln q_i = c$, replacing c_i with $c + \ln p_i - \ln q_i$ in the above equation allows us to deduce

$$\big(\; G, G', c + \ln(p_i \cdot \gamma_i(G, G')) - \ln(q_i \cdot \gamma_i'(G, G')) \; \big) \in \preccurlyeq^\epsilon . \tag{5.3}$$

Let $\mu' = \sum\!\!\!\!\!\!\int_{i\in I} q_i \mu_i'$. We shall construct desirable weight functions π and π' for μ and μ' out of p_i, q_i, γ_i and γ_i'.

Let

$$\begin{aligned} \pi(G, G') &= \textstyle\sum_{i\in I} p_i \gamma_i(G, G'), \\ \pi'(G, G') &= \textstyle\sum_{i\in I} q_i \gamma_i'(G, G'). \end{aligned}$$

Then we have

$$\begin{aligned}
&\textstyle\sum_{G'}\pi(G,G') \\
=&\textstyle\sum_{i\in I}p_i\cdot\sum_{G'}\gamma_i(G,G') && \text{Commutativity}\\
=&\textstyle\sum_{i\in I}p_i\cdot\mu_i(G) && \text{Def. of } \gamma_i\\
=&\mu(G) && \text{Def. of } \mu
\end{aligned}$$

Analogously, we can show $\sum_G \pi'(G,G') = \mu'(G')$. Furthermore,

$$\min_{i\in I}\ln\frac{p_i\gamma_i(G,G')}{q_i\gamma_i'(G,G')} \le \ln\frac{\pi(G,G')}{\pi'(G,G')} \le \max_{i\in I}\ln\frac{p_i\gamma_i(G,G')}{q_i\gamma_i'(G,G')}.$$

By Lemma 5.3.2 and Eq. (5.3), it follows:

$$\big(\ G,G',c+\ln\pi(G,G')-\ln\pi'(G,G')\ \big)\in\preccurlyeq^{\epsilon}\ .$$

Hence, $\mu\ \mathcal{L}^{\mathcal{A}}(\preccurlyeq^{\epsilon},c)\ \mu'$ which completes the proof.

□

Theorem 5.5.2 (Soundness of $\mathcal{A}_2$)**.** *If* $\mathcal{A}_2 \vdash (\epsilon,c) \rhd Q \prec Q'$ *then* $Q \preceq^{(\epsilon,c)} Q'$.

Proof. The proof proceeds by induction on the length of the derivation for $(\epsilon,c) \rhd Q \prec Q'$, with a case analysis on the last applied axiom or inference rule. Here we show the soundness of axioms $\mathbf{T_1}, \mathbf{T_2}$ and $\mathbf{T_3}$. The remaining reasonings follow straightforwardly from Propositions 5.3.9, 5.3.10 and 5.5.1.

Case $\mathbf{T_1}$ By Prop. 5.3.5 we have $\tau.\Delta(F) \preccurlyeq^{(0,0)} F$, namely, $(\tau.\Delta(F),F,0)\in\preccurlyeq^0$. By directly checking the definition of amortised observational congruence (Def. 5.3.6),

$$a.\big(\bigoplus_{i\in I}p_iE_i\oplus p\ \tau.\Delta(F)\big)\stackrel{a}{\Longrightarrow}\mu \text{ implies } a.\big(\bigoplus_{i\in I}p_iE_i\oplus p\ F\big)\stackrel{a}{\Longrightarrow}\mu',$$

and, trivially, $\mu\ \mathcal{L}^{A}(\preccurlyeq^0,0)\ \mu'$ holds, we obtain the required

$$a.\big(\bigoplus_{i\in I}p_iE_i\oplus p\ \tau.\Delta(F)\big)\preceq^{(0,0)} a.\big(\bigoplus_{i\in I}p_iE_i\oplus p\ F\big).$$

Case $\mathbf{T_2}$ Compared with the right-hand side (RHS) of the inequality, the left-hand side (LHS) is enriched with just one summand $a.\bigoplus_{i\in I} p_i P_i$. Thus we only need to show that if the LHS proceeds any weak transitions in $a.\bigoplus_{i\in I} p_i P_i$, the RHS has corresponding transitions to match with them w.r.t. $\preceq^{(0,0)}$.

Assume that $a.\bigoplus_{i\in I} p_i P_i \stackrel{a}{\Longrightarrow} \mu$, in which $\mu = \sum_{i\in I} p_i \mu_i$ where $P_i \mapsto \mu_i$. We aim to find also a weak transition from the RHS such that $\tau.\bigoplus_{i\in I} p_i(E_i + a.P_i) \stackrel{a}{\Longrightarrow} \mu$. Then it allows us to obtain easily that $\mu\ \mathcal{L}^A(\preccurlyeq^0, 0)\ \mu$ holds.

We have the following derivations:

$$\frac{\forall i \in I, \dfrac{\dfrac{P_i \mapsto \mu_i}{P_i \Rightarrow \mu_i}(1)}{E_i + a.P_i \stackrel{a}{\Longrightarrow} \mu_i}(2) \quad \tau.\bigoplus_{i\in I} p_i(E_i + a.P_i) \Rightarrow \sum_{i\in I} p_i(E_i + a.P_i)}{\tau.\bigoplus_{i\in I} p_i(E_i + a.P_i) \stackrel{a}{\Longrightarrow} \sum_{i\in I} p_i\mu_i}(3)$$

where (1) is obtained from **p-idle** and **Weak2**; (2) is from **Weak1**, **prefix** and **rchoice**, and (3) is from **Weak5**. Hence we obtain $\tau.\bigoplus_{i\in I} p_i (E_i + a.P_i) \stackrel{a}{\Longrightarrow} \mu$ as required.

Case $\mathbf{T_3}$ Similarly to the above case, we only need to show that, for any weak transition performed by the LHS of the inequality, the RHS has a matching transition.

Assume that $a.\bigoplus_{i\in I\cup J} p_i P_i \stackrel{a}{\Longrightarrow} \mu$, in which $\mu = \sum_{i\in I\cup J} p_i \mu_i$ where $P_i \mapsto \mu_i$. We aim to find also a weak transition from the RHS such that $a.\big(\bigoplus_{i\in I} p_i(E_i + \tau.P_i) \oplus \bigoplus_{j\in J} p_j E_j\big) \stackrel{a}{\Longrightarrow} \mu$.

We have the following derivations:

$$\forall i \in I \frac{\dfrac{P_i \mapsto \mu_i}{P_i \Rightarrow \mu_i}(1')}{E_i + \tau.P_i \Rightarrow \mu_i}(2'),$$

where $(1'), (2')$ are obtained in the same way as the foregoing $(1), (2)$ in Case $\mathbf{T_2}$.

For $j \in J$, by **Weak3**, $E_j \Longrightarrow \delta(E_j)$, by $P_j = \Delta(E_j) \mapsto \mu_j$ we have $E_j \Longrightarrow \mu_j$.

Trivially,

$$a.\big(\bigoplus_{i\in I} p_i(E_i + \tau.P_i) \oplus \bigoplus_{j\in J} p_j E_j\big) \stackrel{a}{\Longrightarrow} \bigoplus_{i\in I} p_i(E_i + \tau.P_i) \oplus \bigoplus_{j\in J} p_j E_j.$$

By **Weak4**, it holds:

$$a.\big(\bigoplus_{i\in I} p_i(E_i + \tau.P_i) \oplus \bigoplus_{j\in J} p_j E_j\big) \stackrel{a}{\Longrightarrow} \sum_{i\in I\cup J} p_i\mu_i.$$

Hence, we obtain $a.\big(\bigoplus_{i\in I} p_i(E_i + \tau.P_i) \oplus \bigoplus_{j\in J} p_j E_j\big) \stackrel{a}{\Longrightarrow} \mu$ as required.

□

Now we turn to the proof of the completeness result. As usual we need to use a stronger version of a normal form.

Definition 5.5.3. *A process Q is in* full normal form*(FNF) if*

- *either $Q \equiv \mathbf{0}$;*
- *or $Q \equiv \sum_{i\in I} a_i.\bigoplus_{j\in J_i} p^i_j E^i_j$, where each E^i_j is also in full normal form;*
- *If $Q \stackrel{a}{\Longrightarrow} \mu$, then $Q \stackrel{a}{\longrightarrow} \mu$.*

Lemma 5.5.4. *(saturation) If $Q \stackrel{a}{\Longrightarrow} \mu$, then $\mathcal{A}_2 \vdash (0,0) \rhd Q \prec Q + a.P$ where $P \mapsto \mu$.*

Proof. We prove by induction on the depth $d(Q)$ of Q.

The case when $d(Q) = 0$ is trivial.

For the inductive step, three cases need to be considered. Let $P \mapsto \mu$.

Case 1 If $a.P$ is a summand of Q, then the conclusion holds by A′2, **Symm** and **Triangle**.

Case 2 If $a.P'$ is a summand of Q, assume that $P' \equiv \bigoplus_{i\in I} p_i E_i \oplus \bigoplus_{j\in J} p_j E_j$ where for each $i \in I$, $E_i \stackrel{\tau}{\Longrightarrow} \mu_i$, and for each $j \in J$, E_j stays unchanged, namely $E_j \Rightarrow \mu_j$ where $\mu_j = \delta(E_j)$. Let $\sum_{i\in I\cup J} p_i\mu_i = \mu$. By the induction hypothesis, for $i \in I$, $\mathcal{A}_2 \vdash (0,0) \rhd E_i \prec E_i + \tau.P_i$ where $P_i \mapsto \mu_i$. For $j \in J$ let $P_j \mapsto \mu_j$, so $\mathcal{A}_2 \vdash (0,0) \rhd$

$$
\begin{aligned}
Q &\prec Q + a.P' && \text{by A}'2\text{, } \textbf{Symm} \text{ and } \textbf{Triangle}\\
&\prec Q + a.\big(\textstyle\bigoplus_{i\in I} p_i E_i \oplus \bigoplus_{j\in J} p_j E_j\big)\\
&\prec Q + a.\big(\textstyle\bigoplus_{i\in I} p_i (E_i + \tau.P_i) \oplus \bigoplus_{j\in J} p_j E_j\big) && \text{by IH and } \textbf{Subs}\\
&\prec Q + a.\big(\textstyle\bigoplus_{i\in I} p_i (E_i + \tau.P_i) \oplus \bigoplus_{j\in J} p_j E_j\big) + a.\textstyle\bigoplus_{i\in I\cup J} p_i P_i\\
& && \text{by } \mathbf{T_3}\text{, } \textbf{Symm} \text{ and } \textbf{Triangle}\\
&\prec Q + a.\textstyle\bigoplus_{i\in I\cup J} p_i P_i && \text{by reversing previous steps}\\
&\prec Q + a.P && \text{by } P \mapsto \mu, \forall i \in I \cup J.\ P_i \mapsto \mu_i \text{ and } \textstyle\sum_{i\in I\cup J} p_i\mu_i = \mu.
\end{aligned}
$$

Case 3 If $\tau.P'$ is a summand of Q, assume that $P' \equiv \bigoplus_{i\in I} p_i E_i$ where for each i, $E_i \stackrel{a}{\Longrightarrow} \mu_i$ and $\sum_{i\in I} p_i\mu_i = \mu$. Then by induction $\mathcal{A}_2 \vdash (0,0) \rhd E_i \prec E_i + a.P_i$ where $P_i \mapsto \mu_i$, so $\mathcal{A}_2 \vdash (0,0) \rhd$

$$
\begin{aligned}
Q &\prec Q + \tau.P' && \text{by A}'2\text{, } \textbf{Symm} \text{ and } \textbf{Triangle}\\
&\prec Q + \tau.\textstyle\bigoplus_{i\in I} p_i E_i\\
&\prec Q + \tau.\textstyle\bigoplus_{i\in I} p_i (E_i + a.P_i) && \text{by IH and } \textbf{Subs}\\
&\prec Q + \tau.\textstyle\bigoplus_{i\in I} p_i (E_i + a.P_i) + a.\textstyle\bigoplus_{i\in I} p_i P_i\\
& && \text{by } \mathbf{T_2}\text{, } \textbf{Symm} \text{ and } \textbf{Triangle}\\
&\prec Q + a.\textstyle\bigoplus_{i\in I} p_i P_i && \text{by reversing previous steps}\\
&\prec Q + a.P && \text{by } P \mapsto \mu, \forall i \in I.\ P_i \mapsto \mu_i \text{ and } \textstyle\sum_{i\in I} p_i\mu_i = \mu.
\end{aligned}
$$

This completes the proof. □

With the help of Lemma 5.5.4, we can further convert a normal form to a full normal form:

Lemma 5.5.5. *For any normal form Q there is a full normal form Q' of equal depth, such that $\mathcal{A}_2 \vdash (0,0) \rhd Q \prec Q'$.*

Proof. The proof proceeds by induction on the depth of Q, essentially following the same idea as Lemma 17 in Chapter 7.4 of [Mil89]. □

Analogous to the strong case, we can show that full normal forms and their original processes share the same indices.

Lemma 5.5.6. *$Q \preceq^{(\epsilon,c)} Q'$ iff $\widehat{Q} \preceq^{(\epsilon,c)} \widehat{Q'}$, where $\widehat{Q}$ and $\widehat{Q'}$ are full normal forms of Q and Q' respectively.*

Now we are in a position to prove the completeness of $\mathcal{A}_2$.

Theorem 5.5.7 (Completeness of $\mathcal{A}_2$)**.** *If $Q \preceq^{(\epsilon,c)} Q'$ then $\mathcal{A}_2 \vdash (\epsilon, c) \rhd Q \prec Q'$.*

Proof. By Lemmas 5.5.5 and 5.5.6, we may assume that Q and Q' are in full normal form. The proof proceeds by induction on the sum of the depths of Q and Q'.

If $d(Q) = d(Q') = 0$, then $Q \equiv \mathbf{0} \equiv Q'$, so the result is trivial.

Otherwise, assume that $Q \preceq^{(\epsilon,c)} Q'$. Let $a.P$ be a summand of Q. We aim to prove that Q' has a summand provably equal to $a.P$. Now $Q \xrightarrow{a} \mu$ where $P \mapsto \mu$, so there is a P' such that $P' \mapsto \mu'$, $Q' \stackrel{a}{\Longrightarrow} \mu'$ and $\mu \; \mathcal{L}^A(\preccurlyeq^\epsilon, c) \; \mu'$. Moreover $Q' \xrightarrow{a} \mu'$ since Q' is a full normal form, so $a.P'$ is a summand of Q'.

By $\mu \; \mathcal{L}^A(\preccurlyeq^\epsilon, c) \; \mu'$, there exist weight function ω and ω' satisfying $\sum_F \omega(E, F) = \mu(E)$, $\sum_E \omega'(E, F) = \mu'(F)$ and if $\omega(E, F) > 0$,

$$E \preccurlyeq^{(\epsilon, c+\ln\omega(E,F)-\ln\omega'(E,F))} F.$$

Here we cannot yet use the induction hypothesis. But by Proposition 5.3.8, we know that either

$$E \preceq^{(\epsilon, c+\ln\omega(E,F)-\ln\omega'(E,F))} F, \tag{5.4}$$

or

$$\tau.\Delta(E) \preceq^{(\epsilon, c+\ln\omega(E,F)-\ln\omega'(E,F))} F, \tag{5.5}$$

or

$$E \preceq^{(\epsilon, c+\ln\omega(E,F)-\ln\omega'(E,F))} \tau.\Delta(F). \tag{5.6}$$

In the first case (Eq. (5.4)), since E and F are full normal form, and of lesser depth than Q and Q', by induction

$$\mathcal{A}_2 \vdash (\epsilon, c + \ln\omega(E,F) - \ln\omega'(E,F)) \rhd E \prec F.$$

In the second case (Eq. (5.5)) we must first convert $\tau.\Delta(E)$ to full normal form before applying induction. From Lemma 5.5.5, there is a full normal form E', of equal depth to $\tau.\Delta(E)$, such that $\mathcal{A}_2 \vdash (0,0) \rhd \tau.\Delta(E) \prec E'$; but the sum of depths of E' and F is one less than the sum of depths of Q and Q', so by induction we infer that

$$\mathcal{A}_2 \vdash (\epsilon, c + \ln\omega(E,F) - \ln\omega'(E,F)) \rhd E' \prec F,$$

so

$$\mathcal{A}_2 \vdash (\epsilon, c + \ln\omega(E,F) - \ln\omega'(E,F)) \rhd \tau.\Delta(E) \prec F.$$

In the third case (Eq. (5.6)), we can similarly infer that

$$\mathcal{A}_2 \vdash (\epsilon, c + \ln\omega(E,F) - \ln\omega'(E,F)) \rhd E \prec \tau.\Delta(F).$$

Given a process E, we use $\overline{E}$ to denote either simply E itself: $\overline{E} \equiv E$ or a *τ-guarded* E: $\overline{E} \equiv \tau.\Delta(E)$. Applying first **Prob** to combine the sets $\{\ \overline{E} \mid E \in \mathit{supp}(\mu)\ \}$ and $\{\ \overline{F} \mid F \in \mathit{supp}(\mu')\ \}$, and then $\mathbf{T_1}$ to simplify all τ-guarded processes, A′6-7 to combine duplicate probabilistic summands, we obtain that

$$\mathcal{A}_2 \vdash (\epsilon, c) \rhd P \prec P',$$

furthermore by **Prefix** we get that

$$\mathcal{A}_2 \vdash (\epsilon, c) \rhd a.P \prec a.P' \ .$$

Thus, we have shown that from $\mathcal{A}_2$ each summand $a.P$ of Q can be proved equal to a summand of Q'. Similarly each summand $a'.P'$ of Q' can be proved equal to a summand of Q. Finally, using Rules A′2, **Symm** and **Triangle** to eliminate duplicate summands, we conclude that $\mathcal{A}_2 \vdash (\epsilon, c) \rhd Q \prec Q'$. □

5.6 Conclusion

In this chapter, we have presented a weak version of the amortised bisimulation introduced in Section 4.2, Chapter 4, formulated proof systems for the amortised strong bisimulation and observational congruence, and proved their soundness and completeness.

Six

Generalized Bisimulation Metrics

Originally proposed in the seminal works of van Breugel and Worrel [vBW01b, vBW01a] and of Desharnais et al. [DGJP99, DJGP02, DJGP04], the pseudometric based on the Kantorovich lifting has become very popular in the process algebra community. One reason for its success is that, when dealing with probabilistic processes, distances are more suitable than equivalences, since the latter are not robust wrt small variation of probabilities. Another important reason is that, thanks to the dual presentation of the Kantorovich lifting in terms of the mass transportation problem, the distance can be efficiently computed using linear programming algorithms [vBW01a, vBW06, vBW14, BBLM13b]. Furthermore, this pseudometric is an extension of probabilistic bisimilarity, in the sense that two states have distance distance 0 if and only if they are bisimilar. In fact, this pseudometric also shares with bisimilarity a similar coinductive definition. More precisely, it is defined as the greatest fixpoint of a transformation that has the same structure as the one used for bisimilarity.[1] This allows to transfer some of the concepts and methods that have been extensively explored in process algebra, and to use lines of reasoning which the process

[1] In the original definition the Kantorovich bisimilarity pseudometric was defined as the greatest fixpoint, but such definition requires using the reverse order on metrics. More recently, authors tend to use the natural order, and define the bisimilarity metric as the least fixpoint, see [BBLM13a, BBLM13b, CvBW12]. Here we follow the latter approach.

algebra community is familiar with. Along the same lines, a nice property of this pseudometric is that the standard operators of process algebra are non-expansive w.r.t. it. This generalizes the result that bisimulation is a congruence, and can be used in a similar way, for compositional reasoning and verification.

Last but not least, the Kantorovich bisimilarity metric provides a bound on the corresponding distance on probabilistic traces [CvBW12] (corresponding in the sense that the definition is based on the same Kantorovich lifting). This means that it can be used to verify certain probabilistic properties on traces. More specifically, it can be used to verify properties that are expressed in terms of difference between probabilities of sets of traces. These properties are linear, in the sense that the difference increases linearly wrt variations on the distributions.

Many properties, however, such as several privacy and security ones, are not linear. This is the case of the popular property of differential privacy [Dwo06], which is expressed in terms of ratios of probabilities. In fact, there are processes that have small Kantorovich distance, and which are not ϵ-differentially private for any finite ϵ. Another example are the properties used in quantitative information flow, which involve logarithmic functions on probabilities.

The purpose of this work is to generalize the Kantorovich lifting to obtain a family of pseudometrics suitable for the verification of a wide class of properties, following the principles that:

i. the members of this family should depend on a parameter related to the class of properties (on traces) that we wish to verify,

ii. each member should provide a bound on the corresponding distance on trace distributions,

iii. the kernel of each member should correspond to probabilistic bisimilarity,

iv. the general construction should be coinductive,

v. the typical process-algebra operators should be non-expansive,

vi. each member should be feasible to compute.

In this chapter we have achieved the first four desiderata. Regarding the last two, so far we have studied a particular case (hereafter called multiplicative variant of the Kantorovich lifting) based on the notion of distance used in the definition of differential privacy. We were able to find a dual form of the lifting, which allows to reduce the problem of its computation to a linear optimization problem solvable with standard algorithms. We have also proved that several typical process-algebra operators are non-expansive, and we have given explicitly the expression of the bound. For some of them we were able to prove this result in a general form, i.e., non-expansiveness wrt all the metrics of the family, and with the bound represented by the same expression.

As an example of application of our framework, we show how to instantiate our construction to obtain the multiplicative variant of the Kantorovich pseudometric, and how to use it to verify the property of differential privacy.

Related Work Bisimulation metrics based on the standard Kantorovich distance have been used in various applications, such as systems biology [TK10], games [CdAMR08], planning [CP12] and security [CG09]. We consider in this chapter discrete state spaces. Bisimulation metrics on uncountable state spaces have been explored in [DJGP04, FPP05, FPP11]. We define bisimulation metrics as fixed point of an appropriate transformation. Alternative characterizations were provided in terms of coalgebras [vBW01b, vBW05] and real-valued modal logics [DGJP99, DJGP04].

6.1 Preliminaries

The *ball*, w.r.t. a metric $m : X^2 \to [0, +\infty)$, of radius r centered at $x \in X$ is defined as $B^m_r(x) = \{x' \in X : m(x, x') \leq r\}$. A point $x \in X$ is called *isolated* iff there exists $r > 0$ such that $B^m_r(x) = \{x\}$; m is called *discrete* if all points are isolated. The *diameter* (wrt m) of $A \subseteq X$ is defined as $\mathrm{diam}_m(A) = \sup_{x,x' \in A} m(x, x')$. A *geodesic* is a curve on which paths have

minimum distance, i.e. a curve $\gamma : I \to X$, where I is an interval of reals, such that $m(\gamma(a), \gamma(b)) = |a - b|$ for all $a, b \in I$. The *kernel* $\ker(m)$ is an equivalence relation on X defined as

$$(x, x') \in \ker(m) \quad \text{iff } m(x, x') = 0$$

6.2 A general family of Kantorovich liftings

We introduce here a family of liftings from pseudometrics on a set X to pseudometrics on $Prob(X)$. This family is obtained as a generalization of the Kantorovich lifting, in which the Lipschitz condition plays a central role.

Definition 6.2.1. *Given two pseudometric spaces $(X, m), (Y, d_Y)$, we say that $f : X \to Y$ is* 1-Lipschitz *wrt m, d_Y iff $d_Y(f(x), f(x')) \leq m(x, x')$ for all $x, x' \in X$.*

We denote by 1-Lip$[(X, m), (Y, d_Y)]$ the set of all such functions.

A function $f : X \to \mathbb{R}$ can be lifted to a function $\hat{f} : Prob(X) \to \mathbb{R}$ by taking its expected value. For discrete distributions (countable X) it can be written as:

$$\hat{f}(\mu) = \textstyle\sum_{x \in X} \mu(x) f(x) \tag{6.1}$$

while for continuous distributions we need to restrict f to be measurable wrt the corresponding σ-algebra on X, and take $\hat{f}(\mu) = \int f d\mu$.

Given a pseudometric m on X, the *standard Kantorovich lifting* of m is a pseudometric $K(m)$ on $Prob(X)$, defined as:

$$K(m)(\mu, \mu') = \sup\{|\hat{f}(\mu) - \hat{f}(\mu')| : \ f \in \text{1-Lip}[(X, m), (\mathbb{R}, d_{\mathbb{R}})]\}$$

where $d_{\mathbb{R}}$ denotes the standard metric on reals. For continuous distributions we implicitly take the sup to range over measurable functions.

Generalization. A generalization of the Kantorovich lifting can be naturally obtained by extending the range of f from $(\mathbb{R}, d_{\mathbb{R}})$ to a generic met-

ric space (V, d_V), where $V \subseteq \mathbb{R}$ is a convex subset of the reals[2], and d_V is a metric on V. A function $f : X \to V$ can be lifted to a function $\hat{f} : Prob(X) \to V$ in the same way as before (cfr. (6.1)); the requirement that V is convex ensures that $\hat{f}(\mu) \in V$.

Then, similarly to the standard case:

Definition 6.2.2. *Given a pseudometric space (X, m), we can define a lifted pseudometric $K_V(m)$ on $Prob(X)$ as:*

$$K_V(m)(\mu, \mu') = \sup\{d_V(\hat{f}(\mu), \hat{f}(\mu')) : \ f \in 1\text{-}Lip[(X, m)(V, d_V)]\}$$

The subscript V in K_V is to emphasize the fact that for each choice of (V, d_V) we may get a different lifting. We should also point out the difference between m, the pseudometric on X being lifted, and d_V, the metric (not pseudo) on V which parameterizes the lifting.

The constructed $K_V(m)$ can be shown to be an extended pseudometric for any choice of (V, d_V), i.e. it is non-negative, symmetric, identical elements have distance zero, and it satisfies the triangle inequality. However, without extra conditions, it is not guaranteed to be bounded (even if m itself is bounded). For the purposes of this chapter this is not an issue. Below we show that under the condition that d_V is *ball-convex* (i.e. all its balls are convex sets, which holds for all metrics in this chapter), the following bound can be obtained:

$$K_V(m)(\mu, \mu') \leq \mathrm{diam}_m(supp(\mu) \cup supp(\mu'))$$

We say that (V, d_V) is *ball-convex* if $B_r^{d_V}(x)$ is convex for all $r > 0, x \in V$. Not all metrics have this property, in fact in [Nor91] it is shown that (V, d_V) is ball-convex iff

$$d_V(x, \lambda y_1 + \bar{\lambda} y_2) \leq \max\{d_V(x, y_1), d_V(x, y_2)\} \qquad \forall x, y_1, y_2 \in V, \lambda \in [0, 1]$$

i.e. iff $d_V(x, \cdot)$ is a quasi-convex function for any fixed $x \in V$. Many standard metrics (for instance all norms) satisfy this property. Moreover the metric

[2] V could be further generalized to be a convex subset of a vector space. It is unclear whether such a generalization would be useful, hence it is left as future work.

$d_\otimes$ used in the multiplicative Kantorovich variant (Section 6.4) also satisfies it.

The usefulness of ball-convexity is given by the following proposition, stating that on such metrics, convex combinations cannot increase distances. We denote by $\mathrm{ch}(A)$ the convex hull of A.

Proposition 6.2.3. *Let (V, d_V) be ball-convex and $A \subseteq V$. Then*

$$\mathrm{diam}_{d_V}(\mathrm{ch}(A)) = \mathrm{diam}_{d_V}(A)$$

Proof. From $A \subseteq \mathrm{ch}(A)$ we get $\mathrm{diam}_{d_V}(A) \leq \mathrm{diam}_{d_V}(\mathrm{ch}(A))$. We now show that $\mathrm{diam}_{d_V}(\mathrm{ch}(A)) \leq \mathrm{diam}_{d_V}(A)$.

Let $\delta = \mathrm{diam}_{d_V}(A)$ and assume that $\mathrm{diam}_{d_V}(\mathrm{ch}(A)) > \delta$, i.e. $\exists x, y \in \mathrm{ch}(A)$ s.t. $d_V(x, y) > \delta$. If $A \subseteq B_\delta^{d_V}(x)$ then $\mathrm{ch}(A) \subseteq \mathrm{ch}(B_\delta^{d_V}(x)) = B_\delta^{d_V}(x)$ (balls are convex) which is a contradiction since $y \notin B_\delta^{d_V}(x)$. Hence it must hold that $A \not\subseteq B_\delta^{d_V}(x)$, that is $\exists z \in A$ with $d_V(x, z) > \delta$.

Assume that $A \subseteq B_\delta^{d_V}(z)$. Then $\mathrm{ch}(A) \subseteq \mathrm{ch}(B_\delta^{d_V}(z)) = B_\delta^{d_V}(z)$. Since $x \in \mathrm{ch}(A)$, $x \in B_\delta^{d_V}(z)$ which contradicts $d_V(x, z) > \delta$. Hence $A \not\subseteq B_\delta^{d_V}(z)$. Therefore, $d(w, z) > \delta$ for some $w \in A$. This contradicts our assumption that $\mathrm{diam}_{d_V}(A) = \delta$. □

As a corollary of the previous result, we can bound the Kantorovich lifting of a pseudometric m.

Proposition 6.2.4. *Let (V, d_V) be ball-convex. Then*

$$K_V(m)(\mu, \mu') \leq \mathrm{diam}_m(\mathit{supp}(\mu) \cup \mathit{supp}(\mu'))$$

Proof. Let $f \in 1\text{-Lip}[(X, m)(V, d_V)]$, let $A = \mathit{supp}(\mu) \cup \mathit{supp}(\mu')$, and let $f(A)$ denote the set $\{f(x) : x \in A\}$. We have that

$$\begin{aligned} d_V(\hat{f}(\mu), \hat{f}(\mu')) &\leq \mathrm{diam}_{d_V}(\mathrm{ch}(f(A))) && \hat{f}(\mu), \hat{f}(\mu') \in \mathrm{ch}(f(A)) \\ &= \mathrm{diam}_{d_V}(f(A)) && \text{Prop 6.2.3} \\ &\leq \mathrm{diam}_m(A) && \text{1-Lipschitz} \end{aligned}$$

This holds for all 1-Lipschitz functions, hence $K_V(m)(\mu, \mu') \leq \mathrm{diam}_m(A)$. □

Examples The standard Kantorovich lifting is obtained by taking $(V, d_V) = (\mathbb{R}, d_{\mathbb{R}})$. When 1-bounded pseudometrics are used, like in the construction of the standard bisimilarity metric, then we can equivalently take $V = [0, 1]$.

Moreover, a multiplicative variant of the Kantorovich lifting can be obtained by taking $(V, d_V) = ([0, 1], d_{\otimes})$ (or equivalently $([0, \infty), d_{\otimes})$) where $d_{\otimes}(x, y) = |\ln x - \ln y|$. The resulting lifting is discussed in detail in Section 6.4 and its relation to differential privacy is shown in Section 6.4.2.

6.3 A general family of bisimilarity pseudometrics

In this section we define a general family of pseudometrics on the states of a PA which have the property of extending probabilistic bisimilarity in the usual sense. Following standard lines, we define a transformation on state pseudometrics by first lifting a state pseudometric to a pseudometric on distributions (over states), using the generalized Kantorovich lifting defined in previous section. Then we apply the standard Hausdorff lifting to obtain a pseudometric on sets of distributions. This last step is to take into account the nondeterminism of the PA, i.e., the fact that in general, from a state, we can make transitions to different distributions. The resulting pseudometric naturally corresponds to a state pseudometric, obtained by associating each set of distributions to the states which originate them. Finally, we define the intended bisimilarity pseudometric as the least fixpoint of this transformation wrt the ordering $\preceq$ on the state pseudometrics (or equivalently, as the greatest fixpoint wrt the reverse of $\preceq$). We recall that $m \preceq m'$ means that $m(s, s') \leq m'(s, s')$ for all $s, s' \in S$.

Let $\mathcal{A} = (S, \overline{s}, A, D)$ be a PA, assume that $\mathcal{A}$ is finitely branching. Let (V, d_V) be a metric space (for some convex $V \subseteq \mathbb{R}$), and let $\mathcal{M}$ be the set of pseudometrics m on S such that $\mathrm{diam}_m(S) \leq \mathrm{diam}_{d_V}(V)$. Recall that $\inf \emptyset = \mathrm{diam}_{d_V}(V)$ and $\sup \emptyset = 0$.

Definition 6.3.1. *The transformation $F_V : \mathcal{M} \to \mathcal{M}$ is defined as follows.*

$$F_V(m)(s,t) = \max_{a \in A}\{ \sup_{s \xrightarrow{a} \mu} \inf_{t \xrightarrow{a} \nu} K_V(m)(\mu,\nu), \sup_{t \xrightarrow{a} \nu} \inf_{s \xrightarrow{a} \mu} K_V(m)(\nu,\mu)\}$$

We can also characterize F_V in terms of the following zigzag formulation:

Proposition 6.3.2. *For any $\epsilon \geq 0$, $F_V(m)(s,t) \leq \epsilon$ if and only if:*

- *if $s \xrightarrow{a} \mu$, then there exists ν such that $t \xrightarrow{a} \nu$ and $K_V(m)(\mu,\nu) \leq \epsilon$,*
- *if $t \xrightarrow{a} \nu$, then there exists μ such that $s \xrightarrow{a} \mu$ and $K_V(m)(\nu,\mu) \leq \epsilon$.*

Proof. The proposition can be proved by directly checking the definition of F_V. □

The following result states that K_V and F_V are monotonic wrt $(\mathcal{M}, \preceq)$.

Proposition 6.3.3. *Let $m, m' \in \mathcal{M}$. If $m \preceq m'$ then:*

$$F_V(m)(s,s') \leq F_V(m')(s,s') \quad \textit{for all states } s, s'$$
$$K_V(m)(\mu,\mu') \leq K_V(m')(\mu,\mu') \quad \textit{for all distributions } \mu, \mu'$$

Proof. The essence of the proof is the observation that

$$1\text{-Lip}[(V,d_V),(S,m)] \subseteq 1\text{-Lip}[(V,d_V),(S,m')]$$

whenever $m \preceq m'$. □

Since $(\mathcal{M}, \preceq)$ is a complete lattice and F_V is monotone on $\mathcal{M}$, by Tarski's theorem [Tar55] F_V has a least fixpoint, which coincides with the least pre-fixpoint. We define the *bisimilarity pseudometric* bm_V as this least fixpoint:

Definition 6.3.4. *The* bisimilarity pseudometric *bm_V is defined as:*

$$bm_V = \min\{m \in \mathcal{M} \mid F_V(m) = m\} = \min\{m \in \mathcal{M} \mid F_V(m) \preceq m\}$$

In addition, if the states of $\mathcal{A}$ are finite, then the closure ordinal of F_V is ω (cf: [DJGP02], Lemma 3.10). Hence we can approximate bm_V by iterating the function F_V from the bottom element:

Proposition 6.3.5. *Assume that S is finite. Let $m_0 = \bot$ and $m_{i+1} = F_V(m_i)$. Then $bm_V = \sup_i m_i$.*

Proof. Since the closure ordinal of F_V is ω, following the standard way, one can approximate the least fixpoint bm_V by iterating the function F_V from the bottom element. □

The next section shows that bm_V is indeed a bisimilarity metric, in the sense that its kernel coincides with probabilistic bisimilarity.

6.3.1 Bisimilarity as 0-distance

We now show that under certain conditions, the pseudometric constructed from $K_V(m)$ characterizes bisimilarity at its kernel. Recall that the kernel $\ker(m)$ of m is an equivalence relation relating states at distance 0.

To obtain the characterization result we assume that (a) the PA is finitely branching, and (b) there exists a geodesic in (V, d_V). The main result is that, under condition (b), the kernel operator and the lifting operators $\mathcal{L}, K_V$ commute on distributions with finite support.[3] This is then sufficient to obtain the characterization result due to condition (a).

Lemma 6.3.6. *If (V, d_V) has a geodesic then $\mathcal{L}(\ker(m))$ and $\ker(K_V(m))$ coincide on all distributions of finite support.*

Proof. Direction $\subseteq$: let $(\mu, \mu') \in \mathcal{L}(\ker(m))$ and let $f : S \to V$ be 1-Lipschitz wrt m, d_V. Every such function needs to map equivalent elements of S to the same element of V, since $(s, s') \in \ker(m)$ implies $m(s, s') = 0$ which, from 1-Lipschitz, means that $d_V(f(s), f(s')) = 0$ which in turn implies $f(s) = f(s')$.

[3] cfr. [DD09] for the analogous property for the standard Kantorovich lifting.

For simplicity, we write $[s]$ for $[s]_{\ker(m)}$. Let S_r be a set of representatives of each class, i.e. $S = \uplus_{s \in S_r} [s]$. Then

$$\begin{aligned}
\hat{f}(\mu) &= \sum_{s \in S_r} \sum_{s' \in [s]} \mu(s') f(s') \\
&= \sum_{s \in S_r} f(s) \mu([s]) && f(s') \text{ is common for the class} \\
&= \sum_{s \in S_r} f(s) \mu'([s]) && (\mu, \mu') \in \mathcal{L}(\ker(m)) \\
&= \hat{f}(\mu')
\end{aligned}$$

Hence $d_V(\hat{f}(\mu), \hat{f}(\mu')) = 0$ and this happens for all such f, which implies $K_V(m)(\mu, \mu') = 0$, that is $(\mu, \mu') \in \ker(K_V(m))$. Note that this direction requires neither an assumption on (V, d_V), nor that μ, μ' have finite support.

Direction $\supseteq$: let $(\mu, \mu') \notin \mathcal{L}(\ker(m))$ such that $S_+ = supp(\mu) \cup supp(\mu')$ is finite; we show that $(\mu, \mu') \notin \ker(K_V(m))$. Since μ, μ' are not equivalent, there exists $s_0 \in S$ such that $\mu([s_0]) \neq \mu'([s_0])$. Let $\zeta > 0$ be the minimum distance between s_0 and elements of S_+ not equivalent to s_0, that is

$$\zeta = \min_{s \in S_+ \setminus [s_0]} m(s, s_0)$$

Moreover, let $\gamma : [0, d] \to V$ be a geodesic[4] of (V, d_V), and take some $\epsilon > 0$ that is smaller than both ζ and d.

We define a function $f : S \to V$ as:

$$f = \gamma \circ g \qquad \text{where} \qquad g(s) = \min\{m(s, s_0), \epsilon\}$$

We first show that f is 1-Lipshitz wrt m, d_V. Let $s, s' \in S$ and assume wlog that $g(s) \geq g(s')$. From the definiton of g it follows that:

$$g(s) - g(s') \leq m(s, s_0) - m(s', s_0) \tag{6.2}$$

Then we have that:

$$\begin{aligned}
d_V(f(s), f(s')) &= d_V(\gamma(g(s)), \gamma(g(s'))) && \text{Def. of } f \\
&= g(s) - g(s') && \gamma \text{ is a geodesic} \\
&\leq m(s, s_0) - m(s', s_0) && (6.2) \\
&\leq m(s, s') && \text{triangle ineq.}
\end{aligned}$$

[4] Wlog we can take γ's domain to be of the form $[0, d]$.

hence f is 1-Lipshitz wrt m, d_V.

Moreover, since $\epsilon < \zeta$, for all elements $s \in S_+$ we have that either $g(s) = 0$ (when $s \in [s_0]$) or $g(s) = \epsilon$, hence f maps all elements of $S_+ \cap [s_0]$ to $\gamma(0)$ and all elements of $S_+ \setminus [s_0]$ to $\gamma(\epsilon)$. Finally, for any $a \neq b \in \mathbb{R}, \lambda \neq \lambda' \in [0,1]$ it holds that $a\lambda + b(1-\lambda) \neq a\lambda' + b(1-\lambda')$, as a consequence:

$$\begin{aligned}\hat{f}(\mu) &= \textstyle\sum_{s \in S_+} \mu(s) f(s) \\ &= \gamma(0)\mu\left([s_0]\right) + \gamma(\epsilon)(1 - \mu\left([s_0]\right)) \\ &\neq \gamma(0)\mu'([s_0]) + \gamma(\epsilon)(1 - \mu'([s_0])) \\ &= \hat{f}(\mu')\end{aligned}$$

Hence $d_V(\hat{f}(\mu), \hat{f}(\mu')) > 0$ which implies $K_V(m)(\mu, \mu') > 0$, that is $(\mu, \mu') \notin \ker(K_V(m))$. □

Note that in the above proof we need a geodesic γ since in general there might be elements of S arbitrarily close to s_0, and we need to map such elements to V while preserving the 1-Lipshitz condition. However, if S is finite, we can always find an $\epsilon > 0$ smaller than the distance between s_0 and any $s \notin [s_0]$. In this case it is enough that (V, d_V) has a *non-isolated point* a, so we can find $b \in V$ s.t. $d_V(a, b) < \epsilon$, then define f as $f(s) = a$ iff $s \in [s_0]$ and $f(s) = b$ otherwise, and continue the proof in the same way.

If S is finite, the same result can be obtained under the weaker condition that (V, d_V) is non-discrete. We also expect the result to be extensible to distributions with infinite support.

We now show the correspondence between pre-fixpoint metrics and bisimulations. Using Lemma 6.3.6, we can see that the definition of the transformation B for bisimulations in Chapter 2.4 corresponds to the characterization of F_V in Proposition 6.3.2, for $\epsilon = 0$. Hence we have the follow proposition hold. Note that here distributions are assumed to have finite support sets.

Proposition 6.3.7. *Assume that (V, d_V) has a geodesic. For every $m \in \mathcal{M}$, if $F_V(m) \preceq m$ then $\ker(m) \subseteq B(\ker(m))$, i.e., $\ker(m)$ is a bisimulation.*

Proof. Let $(s,t) \in \ker(m)$, i.e. $m(s,t) = 0$. Since $F_V(m) \preceq m$, by Prop. 6.3.2, we have that if $s \xrightarrow{a} \mu$, then there exists ν such that $t \xrightarrow{a} \nu$ and $K_V(m)(\mu,\nu) = 0$. Clearly $(\mu,\nu) \in \ker(K_V(m))$, by Lemma 6.3.6, it follows that $(\mu,\nu) \in \mathcal{L}(\ker(m))$. A similar condition holds for the converse direction where t initiates transitions. Hence, we have $(s,t) \in B(\ker(m))$. □

As a consequence, $\ker(bm_V) \subseteq \sim$. The converse of Proposition 6.3.7 does not hold, because the fact that $\ker(m) \subseteq B(\ker(m))$ does not say anything about the effect of F_V on the distance between elements that are not in the kernel. However, in the case of bisimilarity we can make a connection: consider the greatest metric $m_\sim$ whose kernel coincides with bisimilarity, namely, $m_\sim(s,s') = 0$ if $s \sim s'$ and $m_\sim(s,s') = \mathrm{diam}_{d_V}(V)$ otherwise. We have that $F_V(m_\sim) \preceq m_\sim$, and therefore $\sim = \ker(m_\sim) \subseteq \ker(bm_V)$. Therefore we can conclude that the kernel of the bisimilarity pseudometrics coincides with bisimilarity.

Theorem 6.3.8. *If (V,d_V) has a geodesic, then $\ker(bm_V) = \sim$.*

Proof. Since bm_V is a fixpoint of F_V, then by definition $F_V(bm_V) = bm_V$, by Prop. 6.3.7 $\ker(bm_V) \subseteq B(\ker(bm_V))$, and hence $\ker(bm_V)$ is a probabilistic bisimulation relation, namely, $\ker(bm_V) \subseteq \sim$.

Vice versa, define $m(s,t) = 0$ if $s \sim t$, and $m(s,t) = \mathrm{diam}_{d_V}(V)$ otherwise. Due to Lemma 6.3.6 we have $F_V(m) \preceq m$, hence $bm_V \preceq m$, therefore $\sim = \ker(m) \subseteq \ker(bm_V)$. □

6.3.2 Relation with trace distributions

In this section, we show the relation between the bisimilarity metric bm_V and the corresponding metric on traces, in the case of FPAs (fully probabilistic automata). Note that we restrict to the fully probabilistic case here, where probabilities on traces can defined in the way shown in the preliminaries. The full case of PAs can be treated by using schedulers, but a proper treatment involves imposing scheduler restrictions which complicate the formalism. Since these problems are orthogonal to the goals of this chapter, we keep the discussion simple by restricting to the fully probabilistic case.

The distance between trace distributions (i.e. distributions over A^ω) will be measured by the Kantorovich lifting of the *discrete metric*. Given (V, d_V), let $\delta_V = \mathrm{diam}_{d_V}(V)$. Then let dm_{δ_V} be the δ_V-valued discrete metric on A^ω, defined as $dm_{\delta_V}(\vec{t}, \vec{t'}) = 0$ if $\vec{t} = \vec{t'}$, and $dm_{\delta_V}(\vec{t}, \vec{t'}) = \delta_V$ otherwise.

Then $K_V(dm_{\delta_V})(\mu, \mu')$ is a pseudometric on $Prob(A^\omega)$, whose kernel coincides with probabilistic trace equivalence.

Proposition 6.3.9. *$K_V(dm_{\delta_V})(\mu, \mu') = 0$ iff $\mu(\sigma) = \mu'(\sigma)$ for all measurable $\sigma \subseteq A^\omega$.*

Proof. We have

$$\begin{aligned} & K_V(dm_{\delta_V})(\mu, \mu') = 0 \\ \text{iff} \quad & \text{for any } f \in 1\text{-Lip}[(A^\omega, dm_{\delta_V})(V, d_V)], d_V(\hat{f}(\mu), \hat{f}(\mu')) = 0 \qquad \text{Def. 6.2.2} \\ \text{iff} \quad & \text{for any } f \in 1\text{-Lip}[(A^\omega, dm_{\delta_V})(V, d_V)], \hat{f}(\mu) = \hat{f}(\mu') \qquad d_V \text{ is a metric.} \end{aligned} \tag{6.3}$$

We shall show that the right hand part (6.3) of the above relation is equivalent to the right hand part of Proposition 6.3.9.

($\Leftarrow$) If $\mu(\sigma) = \mu'(\sigma)$ for all measurable $\sigma \subseteq A^\omega$, by checking the definition of $\hat{f}$, it is straightforward that $\hat{f}(\mu) = \hat{f}(\mu')$ for any f.

($\Rightarrow$) For the converse direction, we assume that there exists a measurable $\sigma \subseteq A^\omega$ such that $\mu(\sigma) \neq \mu'(\sigma)$. We construct a non-expansive function $f \in 1\text{-Lip}[(A^\omega, dm_{\delta_V})(V, d_V)]$: $f(\vec{t}) = c$ for $\vec{t} \in \sigma$, 0 otherwise, where c is a constant in V. We get that $\hat{f}(\mu) = c \cdot \mu(\sigma)$ and $\hat{f}(\mu') = c \cdot \mu'(\sigma)$. Due to the assumption, $\hat{f}(\mu) \neq \hat{f}(\mu')$, which contradicts (6.3). □

The following theorem expresses that our bisimilarity metric bm_V is a bound on the distance on traces, which extends the standard relation between probabilistic bisimilarity and probabilistic trace equivalence.

Theorem 6.3.10. *Let $\mu = \Pr[s \rhd \cdot\,]$ and $\mu' = \Pr[s' \rhd \cdot\,]$. Then*

$$K_V(dm_{\delta_V})(\mu, \mu') \leq bm_V(s, s')$$

Proof. For $h \in \mathbb{N}$, we define

$$dm^h_{\delta_V}(\vec{t}, \vec{t'}) = \begin{cases} 0 & \text{if } \vec{t}_h = \vec{t'}_h \\ \delta_V & \text{otherwise,} \end{cases}$$

where $\vec{t}_h$ is the prefix of $\vec{t}$ of length h. The proof of showing $K_V(dm^h_{\delta_V})(\mu, \mu') \leq bm_V(s, s')$ proceeds by induction on h.

For the base case $h = 0$, we have $dm^0_{\delta_V}(\vec{t}, \vec{t'}) = 0$ for any $\vec{t}, \vec{t'}$. Namely, for any $f \in 1\text{-Lip}[(T, dm^0_{\delta_V})(V, d_V)]$, $d_V(f(\vec{t}), f(\vec{t'})) \leq dm^0_{\delta_V}(\vec{t}, \vec{t'}) = 0$. Hence, for any $\vec{t}, \vec{t'}$, $f(\vec{t}) = f(\vec{t'})$, i.e. f is a constant function. Then $d_V(\int_T f d\mu, \int_T f d\mu') = 0$, we have $K_V(dm^0_{\delta_V})(\mu, \mu') = 0$. And trivially $K_V(dm^0_{\delta_V})(\mu, \mu') \leq bm_V(s, s')$ holds.

For inductive case $h + 1$, by definition,

$$K_V(dm^{h+1}_{\delta_V})(\mu, \mu') = \sup \left\{ d_V(\int_T f d\mu, \int_T f d\mu') : f \in 1\text{-Lip}[(T, dm^{h+1}_{\delta_V})(V, d_V)] \right\}$$

Consider the case where both s and s' have no outgoing transitions, then both $K_V(dm^{h+1}_{\delta_V})(\mu, \mu') = 0$ and $bm_V(s, s') = 0$. Consider another case where s can perform a transition with a label that can not be matched by any transition from s', then both $K_V(dm^{h+1}_{\delta_V})(\mu, \mu') = \delta_V$ and $bm_V(s, s') = \delta_V$. Hence, the only interesting case is when $s \xrightarrow{a} \nu$ and $s' \xrightarrow{a} \nu'$.

Let $f \in 1\text{-Lip}[(T, dm^{h+1}_{\delta_V})(V, d_V)]$. The main idea is to show that for any such function f, we can construct a function $g : S \to V$ that is 1-Lipschitz wrt bm_V, d_V.

We define $f_a(\vec{t}) = f(a^\frown \vec{t})$. Clearly $f_a \in 1\text{-Lip}[(T, dm^h_{\delta_V})(V, d_V)]$ and also define $g : S \to V$ as

$$g(s) = \hat{f}_a(\mu) = \int_T f_a d\mu$$

We have that for all $s, s' \in S$

$$\begin{aligned} d_V(g(s), g(s')) &= d_V(\hat{f}_a(\mu), \hat{f}_a(\mu')) && \text{def. of } g(s) \\ &\leq K_V(dm^h_{\delta_V})(\mu, \mu') && f_a \text{ is 1-Lip. wrt } dm^h_{\delta_V}, d_V \\ &\leq bm_V(s, s') && \text{induction hypothesis} \end{aligned}$$

hence g is 1-Lipschitz wrt bm_V, d_V.

Moreover, $\forall \sigma \in \Sigma_T$, i.e., σ is an element of the Σ-algebra on traces. We have $\sigma = \sigma_a \cup X$ s.t. $\sigma_a = a^\frown\sigma', \sigma' \in \Sigma_T$, and X is a set of traces that do not start with a. Then we have, by definition of $\Pr[s \rhd \cdot]$:

$$\begin{aligned} \mu(\sigma) &= \textstyle\sum_i \nu(s_i)\mu_i(\sigma') \\ \mu'(\sigma) &= \textstyle\sum_i \nu'(s_i)\mu_i(\sigma') \end{aligned}$$

where $\mu_i = \Pr[s_i \rhd \cdot]$.

Hence:

$$\begin{aligned} \hat{f}(\mu) &= \int_T f d\mu \\ &= \int_T f_a \sum_i \nu(s_i) d\mu_i && s \xrightarrow{a} \nu \\ &= \sum_i \nu(s_i) \int_T f_a d\mu_i \\ &= \sum_i \nu(s_i) g(s_i) && \text{Def. of } g \\ &= \hat{g}(\nu) \end{aligned}$$

and similarly $\hat{f}(\mu') = \hat{g}(\nu')$. Hence

$$\begin{aligned} d_V(\hat{f}(\mu), \hat{f}(\mu')) &= d_V(\hat{g}(\nu), \hat{g}(\nu')) \\ &\leq K_V(bm_V)(\nu, \nu') && g \text{ is 1-Lipschitz wrt } bm_V, d_V \\ &= bm_V(s, s') && F_V(bm_V) = bm_V \end{aligned}$$

and the bound holds for all f, hence it also holds for $K_V(dm_{\delta_V}^{h+1})(\mu, \mu')$.

Now we only need the condition that K_V is continuous w.r.t. m, i.e., $\forall \epsilon > 0, \exists \delta$: if $\sup_{a,b} |m(a,b) - m'(a,b)| < \delta$, then $\forall \mu, \mu'. |K_V(m)(\mu, \mu') - K_V(m')(\mu, \mu')| < \epsilon$. Hence the bound holds also for $K_V(dm_{\delta_V})(\mu, \mu')$. □

It should be noted that, although the choice of $K_V(dm_{\delta_V})$ as our trace distribution metric might seem arbitrary, this metric is in fact of great interest. In the case of the standard bisimilarity pseudometric, i.e. when $(V, d_V) = ([0,1], d_\mathbb{R})$, this metric is equal to the well-known *total variation* distance (also known as *statistical distance*), defined as $tv(\mu, \mu') =$

$\sup_\sigma |\mu(\sigma) - \mu'(\sigma)|$:

$$K(dm_{\delta_V}) \;=\; tv \tag{6.4}$$

Theorem 6.3.10 reduces to the result of [CvBW12] relating the total variation distance to the bisimilarity pseudometric. Moreover, in the case of the multiplicative pseudometric, discussed in the next section, $K_V(dm_{\delta_V})$ is the same as the multiplicative distance between distributions, discussed in Section 6.4.2, which plays a central role in differential privacy.

6.4 The multiplicative variant

In this section we investigate the multiplicative variant of the Kantorovich pseudometric, obtained by considering as distance d_V the ratio between two numbers instead of their difference. This is the distance used to define differential privacy. We show that this variant has a dual form, which can be used to compute the metric by using linear programming techniques. In the next section, we will show how to use it to verify differential privacy.

Definition 6.4.1. *The multiplicative variant $K_\otimes$ of the Kantorovich lifting is defined as the instantiation of K_V with $([0,1], d_\otimes)$ where $d_\otimes(x,y) = |\ln x - \ln y|$.*

It is well known that the standard Kantorovich metric has a dual form which can be interpreted in terms of *the Transportation Problem*, namely, the lowest total cost of transporting the mass of one distribution μ to the other distribution μ' given the cost (distance) m between locations (in our case, states). The dual form is shown in Fig. 6.1. Note that both the primal and the dual forms are linear optimization problems. The dual form is particularly suitable for computation, via standard linear programming techniques.

For our multiplicative variant, the objective function of the primal form is not a linear expression, hence the linear programming techniques cannot be applied directly. However, since $\ln \hat{f}(\mu) - \ln \hat{f}(\mu') \;=\; \ln {}^{\hat{f}(\mu)}/_{\hat{f}(\mu')}$ and ln is a monotonically increasing function, the primal problem is actually a linear-fractional program. It is known that such kind of program can be converted

	Standard $K(m)(\mu,\mu')$	Multiplicative $K_\otimes(m)(\mu,\mu')$
Primal	$\max_f \lvert\hat{f}(\mu) - \hat{f}(\mu')\rvert$ subject to $\forall s,s'.\ \lvert f(s) - f(s')\rvert \leq m(s,s')$	$\max_f \lvert\ln \hat{f}(\mu) - \ln \hat{f}(\mu')\rvert$ subject to $\forall s,s'.\ \lvert\ln f(s) - \ln f(s')\rvert \leq m(s,s')$
Dual	$\min_\ell \sum_{i,j} \ell_{ij} m(s_i,s_j)$ subject to $\forall i,j.\ \ell_{ij} \geq 0$ $\forall i.\ \sum_j \ell_{ij} = \mu(s_i)$ $\forall j.\ \sum_i \ell_{ij} = \mu'(s_j)$	$\min \ln z$ subject to $\forall i,j.\ \ell_{ij}, r_i \geq 0$ $\forall i.\ \sum_j \ell_{ij} - r_i = \mu(s_i)$ $\forall j.\ \sum_i \ell_{ij} e^{m(s_i,s_j)} - r_j \leq z \cdot \mu'(s_j)$

Figure 6.1: The standard Kantorovich metric and its multiplicative variant.

to an equivalent linear programming problem and then to a dual program. The detailed transformation is shown in the coming subsection. The dual form of the multiplicative variant obtained in this way is shown in Fig. 6.1. (For the sake of simplicity, the figure shows only the dual form of $\ln \hat{f}(\mu) - \ln \hat{f}(\mu')$. The dual form of $\ln \hat{f}(\mu') - \ln \hat{f}(\mu)$ can be obtained by simply switching the roles of μ and μ'.) Hence, the multiplicative pseudometric can be computed by using linear programming techniques.

Finally, note that the curve $\gamma : [0,a] \to [0,1]$, for $a > 0$, defined by $\gamma(t) = e^{-t}$ is a geodesic of $([0,1], d_\otimes)$, since $d_\otimes(\gamma(a),\gamma(b)) = \lvert\ln e^{-a} - \ln e^{-b}\rvert = \lvert a - b\rvert$. Hence, the conditons of Theorem 6.3.8 are satisfied, which means that $bm_\otimes$, i.e. the bisimulation metric constructed by $K_\otimes$, characterizes bisimulation at its kernel.

6.4.1 Transformations of the linear-fractional program

In case S is finite, and since ln is a monotonically increasing function, the multiplicative Kantorovich distance for $\ln \hat{f}(\mu) - \ln \hat{f}(\mu')$ can be computed

by solving the following linear-fractional program:

$$\begin{aligned} \text{maximize} \quad & \frac{\sum_i \mu(s_i)x_i}{\sum_i \mu'(s_i)x_i} \\ \text{subject to:} \quad & \forall i, j.\ x_i \leq e^{m(s_i,s_j)}x_j. \end{aligned}$$

By the constraint, we know that if $\exists i, x_i > 0$ then all the x_i's are positive, and if $\exists i, x_i < 0$ then all the x_i's are negative, namely, they have the same polarity, thus it does not matter whether x_i's are positive or negative. We now show how the program can be converted to a linear one, and then written in dual form.

Following the techniques in [vBW01a], we extend the dimensions of the feasible region by adding new decision variables y_i for $i \in [1, |s|]$. The extension does not affect the optimal value. This is justified by the new constraints ensuring that in fact $x_i = y_i$ for $i \in [1, |s|]$ (because $m(s_i, s_i) = 0$).

$$\begin{aligned} \text{maximize} \quad & \frac{\sum_i \mu(s_i)x_i}{\sum_j \mu'(s_j)y_j} \\ \text{subject to:} \quad & \forall i, j.\ x_i - e^{m(s_i,s_j)}y_j \leq 0 \\ & \forall i,\ y_i - x_i \leq 0. \end{aligned}$$

Now let

$$\alpha_i = \frac{x_i}{\sum_j \mu'(s_j)y_j}, \quad \beta_i = \frac{y_i}{\sum_j \mu'(s_j)y_j}.$$

Let the two constraints be devided by $\sum_j \mu'(s_j)y_j$, the above linear-fractional problem can be transformed to the equivalent linear program.

$$\begin{aligned} \text{maximize} \quad & \textstyle\sum_i \mu(s_i)\alpha_i \\ \text{subject to:} \quad & \forall i, j.\ \alpha_i - e^{m(s_i,s_j)}\beta_j \leq 0 \\ & \forall i,\ \beta_i - \alpha_i \leq 0 \\ & \textstyle\sum_i \mu'(s_i)\beta_i = 1 \\ & \forall i.\ \alpha_i, \beta_i \geq 0. \end{aligned}$$

Then dualizing[5] the above (primal) linear problem yields:

$$\begin{array}{ll} \text{minimize} & z \\ \text{subject to:} & \forall i.\, \sum_j l_{ij} - r_i \geq \mu(s_i) \\ & \forall j.\, \sum_i l_{ij} e^{m(s_i,s_j)} - r_j \leq z \cdot \mu'(s_j) \\ & \forall i,j.\, l_{ij}, r_i \geq 0 \end{array}$$

which is equivalent to the following program where the first kind of constraints becomes an equation:

$$\begin{array}{ll} \text{minimize} & z \\ \text{subject to:} & \forall i.\, \sum_j l_{ij} - r_i = \mu(s_i) \\ & \forall j.\, \sum_i l_{ij} e^{m(s_i,s_j)} - r_j \leq z \cdot \mu'(s_j) \\ & \forall i,j.\, l_{ij}, r_i \geq 0. \end{array}$$

6.4.2 Application to differential privacy

We recall the notion of differential privacy (Def. 2.6.1). Given a set of databases $\mathcal{X}$; two databases $x, x' \in \mathcal{X}$ are *adjacent*, written $x \smile x'$, if they differ in the value of a single individual. A *mechanism* is a function $\mathcal{M} : \mathcal{X} \to Prob(\mathcal{Z})$ where $\mathcal{Z}$ is some set of reported values. Intuitively, $\mathcal{M}(x)$ gives the outcome of the query when applied to database x, which is a probability distribution since noise is added.

Let $tv_\otimes$ be a multiplicative variant of the total variation distance on $Prob(\mathcal{Z})$ (simply called "multiplicative distance" in [Smi08]), defined as:

$$tv_\otimes(\mu, \mu') = \sup_Z |\ln \frac{\mu(Z)}{\mu'(Z)}|$$

Then differential privacy can be rephrased as follows:

A mechanism $\mathcal{M} : \mathcal{X} \to Prob(\mathcal{Z})$ is ϵ-differentially private iff

$$tv_\otimes(\mathcal{M}(x), \mathcal{M}(x')) \leq \epsilon \qquad \forall x \smile x'$$

[5] See [Lah08] for how to take the dual of a linear program.

In our setting, we assume that the mechanism $\mathcal{M}$ is modelled by a FPA, and the result of the mechanism running on x is the trace produced by the execution of the FPA starting from some corresponding state s_x. That is, $\mathcal{Z} = A^\omega$ and

$$\mathcal{M}(x) = \Pr[s_x \rhd \cdot] \tag{6.5}$$

The relation between differential privacy and the multiplicative bisimilarity metric comes from the fact that $tv_\otimes$ can be obtained as the $K_\otimes$ lifting of the discrete metric on A^ω.

Lemma 6.4.2. *Let $a, a', b, b' \in [0,1]$ s.t. $a + b \leq 1, a' + b' \leq 1$, and let*

$$g(x) = d_\otimes(a + bx, a' + b'x)$$

Then $g(x) \leq \max\{g(0), g(1)\}$ for all $x \in [0,1]$.

Proof. Wlog assume $a, a', b, b' > 0$, we can extend to the case 0 by continuity. Define

$$h(x) = \frac{a \ + b\, x}{a' + b'x}$$

The derivative of h is $h'(x) = \frac{a'b - ab'}{(a'+b'x)^2}$, hence h is monotonically increasing when $a'b \geq ab'$ and monotonically decreasing otherwise. This implies that

$$h(x) \leq \max\{h(0), h(1)\} \quad \text{and} \quad h^{-1}(x) \leq \max\{h^{-1}(0), h^{-1}(1)\} \tag{6.6}$$

We have:

$$\begin{aligned}
g(x) &= |\ln h(x)| && \text{Def. of } d_\otimes \\
&= \max\{\ln h(x), \ln h^{-1}(x)\} \\
&\leq \max\{\ln h(0), \ln h(1), \ln h^{-1}(0), \ln h^{-1}(0)\} && (6.6), \text{ monot. of } \ln \\
&= \max\{g(0), g(1)\}
\end{aligned}$$

□

Lemma 6.4.3. *Let $\delta_V = \mathrm{diam}_{d_\otimes}([0,1]) = +\infty$ and let dm_{δ_V} be the discrete metric on A^ω. Then $tv_\otimes = K_\otimes(dm_{\delta_V})$.*

Proof. Any function $f : A^\omega \to [0,1]$ is 1-Lipschitz wrt $d_\otimes, dm_{\delta_V}$, hence

$$K_\otimes(dm_{\delta_V})(\mu,\mu') = \sup_f d_\otimes(\hat{f}(\mu),\hat{f}(\mu'))$$

where f ranges over all measurable functions and $\hat{f}(\mu) = \int f d\mu$. Recall that

$$tv_\otimes(\mu,\mu') = \sup_Z |\ln \frac{\mu(Z)}{\mu(Z')}| = \sup_Z d_\otimes(\mu(Z),\mu'(Z))$$

Let $\mathbf{1}_Z$ be the indicator function, defined as $\mathbf{1}_Z(x) = 1$ iff $x \in Z$ and $\mathbf{1}_Z(x) = 0$ otherwise. We have that $\hat{\mathbf{1}_Z}(\mu) = \mu(Z)$, hence

$$d_\otimes(\hat{\mathbf{1}_Z}(\mu),\hat{\mathbf{1}_Z}(\mu')) = d_\otimes(\mu(Z),\mu'(Z)) \tag{6.7}$$

Direction $\leq$) This is the easy case, since (6.7) implies that every Z in the definition of $tv_\otimes(\mu,\mu')$ can be matched by an f in the definition of $K_\otimes(dm_{\delta_V})(\mu,\mu')$.

Direction $\geq$) A function ϕ is called *simple* if its image $\mathrm{img}(\phi)$ is a finite set. Let Φ be the set of all measurable simple functions from A^ω to $[0,1]$. Any $\phi \in \Phi$ can be represented as $\phi = \sum_{v\in\mathrm{img}(f)} v \cdot \mathbf{1}_{f^{-1}(v)}$ hence $\hat{\phi}(\mu) = \sum_{v\in\mathrm{img}(f)} v \cdot \mu(f^{-1}(a))$. A simple function ϕ is an indicator function iff $\mathrm{img}(\phi) \subseteq \{0,1\}$.

We are going to show that

$$d_\otimes(\hat{\phi}(\mu),\hat{\phi}(\mu')) \leq tv_\otimes(\mu,\mu') \qquad \forall \phi \in \Phi \tag{6.8}$$

The intuition is that we can bound $d_\otimes(\hat{\phi}(\mu),\hat{\phi}(\mu'))$ from above by changing ϕ's values to either 0 or 1. After replacing all values we end up with an indicator function, for which the distance is bounded by $tv_\otimes(\mu,\mu')$ because of (6.7).

Formally, we show (6.8) by induction on $n = |\mathrm{img}(\phi) \setminus \{0,1\}|$, i.e. the (finite) number of ϕ's values that are neither 0 nor 1. For the base case $n = 0$, ϕ is an indicator function and (6.8) follows directly from (6.7). Now assume (6.8) holds for $n \leq k$ and let $\phi \in \Phi$ s.t. $n = k+1$. Then there exists some $v \in \mathrm{img}(\phi)$ s.t. $0 < v < 1$.

Let $\phi_x \in \Phi$ be the function obtained from ϕ after mapping v to x (hence $\phi = \phi_v$). Note that $\hat{\phi_x}(\mu), \hat{\phi_x}(\mu')$ can be written as $a + bx$ and $a' + b'x$

respectively, with a, a', b, b' satisfying the conditions of Lemma 6.4.2. As a consequence we have that

$$d_{\otimes}(\hat{\phi}_v(\mu), \hat{\phi}_v(\mu')) \leq \max\{d_{\otimes}(\hat{\phi}_0(\mu), \hat{\phi}_0(\mu')), d_{\otimes}(\hat{\phi}_1(\mu), \hat{\phi}_1(\mu'))\}$$

From the induction hypothesis both $d_{\otimes}(\hat{\phi}_0(\mu), \hat{\phi}_0(\mu'))$ and $d_{\otimes}(\hat{\phi}_1(\mu), \hat{\phi}_1(\mu'))$ are bounded from above by $tv_{\otimes}(\mu, \mu')$, which concludes the proof of (6.8).

Having shown (6.8), it only remains to extend it to any non-simple measurable $f : A^{\omega} \to [0, 1]$. This comes by approximating f using simple functions: there exist ϕ_n increasing pointwise and converging pointwise to f. From the Monotone Convergence Theorem we have that $\hat{f}(\mu) = \lim_{n\to\infty} \hat{\phi}_n(\mu)$ (see [Che08], Thm 2.4.10 and 3.1.1). We conclude by the continuity of $d_{\otimes}$, since $\lim_{n\to\infty} d_{\otimes}(\hat{\phi}_n(\mu), \hat{\phi}_n(\mu')) = d_{\otimes}(\hat{f}(\mu), \hat{f}(\mu'))$. □

Let $bm_{\otimes}$ be the instantiation of the bisimilarity metric bm_V with $K_{\otimes}$. The above Lemma, together with Theorem 6.3.10, imply the following result, which makes $bm_{\otimes}$ useful to verify differential privacy:

Theorem 6.4.4. *Let $\mathcal{M}$ be the mechanism defined by (6.8), and assume that*

$$bm_{\otimes}(s_x, s_{x'}) \leq \epsilon \qquad \text{for all } x \smile x'$$

Then $\mathcal{M}$ satisfies ϵ-differential privacy.

Proof. We have that $\mathcal{M}(x) = \Pr[s_x \rhd \cdot\,]$ and $\mathcal{M}(x') = \Pr[s_{x'} \rhd \cdot\,]$, hence:

$$\begin{aligned} tv_{\otimes}(\mathcal{M}(x), \mathcal{M}(x')) &= K_{\otimes}(dm_{\delta_V})(\mathcal{M}(x), \mathcal{M}(x')) && \text{Lemma 6.4.3} \\ &\leq bm_{\otimes}(s_x, s_{x'}) && \text{Theorem 6.3.10} \\ &\leq \epsilon && \text{hypothesis} \end{aligned}$$

□

Note that the use of the multiplicative $bm_{\otimes}$ is crucial in the above result. The following example shows that the standard bisimilarity metric bm (generated by the original Kantorovich lifting) may be very different from the level of differential privacy, which is expected, since bm bounds the additive total variation metric (Theorem 6.3.10 and (6.4)) instead of the multiplicative $tv_{\otimes}$.

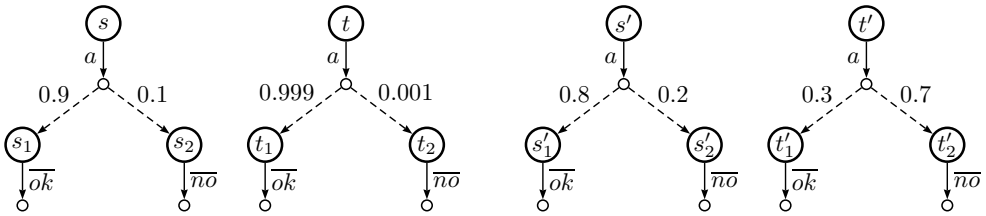

Figure 6.2: The bisimilarity pseudometric bm does not imply differential privacy.

Example 6.4.5. *Consider the processes s, t shown in Fig. 6.2 (a). We have that $bm(s,t) = 0.1 - 0.001 = 0.099$ while their level of differential privacy is $\epsilon = \ln {}^{0.1}/_{0.001} = \ln 100$. Moreover, for the processes s', t' shown in Fig. 6.2 (b) we have $bm(s',t') = 0.7 - 0.2 = 0.5$ while their level of differential privacy is $\epsilon' = \ln {}^{0.7}/_{0.2} = \ln 3.5$. Using the original Kantorovich metric, s and t are considered more indistinguishable than s' and t', in sharp contrast to the corresponding differential privacy levels. That is because the standard Kantorovich metric exhibits an additive nature, which is inadequate for verifying a multiplicative property such as differential privacy.*

Approximate differential privacy. An approximate, also known as (ϵ, δ) version of differential privacy is also widely used [DKM+06], relaxing the definition by an additive factor δ. It requires that:

$$\mathcal{M}(x)(Z) \leq e^{\epsilon}\mathcal{M}(x')(Z) + \delta \qquad \forall x \backsim x', Z \subseteq \mathcal{Z}$$

The α-distance on distributions is proposed in [BKOB12] to capture (ϵ, δ)-differential privacy. For two real numbers a, b and a skew parameter $\alpha \geq 1$, the α-distance between a and b is $\max\{a - \alpha b, b - \alpha a, 0\}$. An instantiation of the Kantorovich lifting based on the α-distance seems promising for extending Theorem 6.4.4 to the approximate case; we leave this extension as future work.

Weak probabilistic anonymity. Weak probabilistic anonymity was proposed in [DPP07] as a measure of the degree of protection of user's identities. It is defined in a way similar to differential privacy, with the crucial

difference (apart from the lack of an adjacency relation) that it uses the (additive) total variation instead of the multiplicative one. Formally, let $\mathcal{X}$ contain the users' identities, and let $\mathcal{M} : \mathcal{X} \to Prob(\mathcal{Z})$ be the system in which users operate. We say that $\mathcal{M}$ is *ϵ-weakly probabilistically anonymous* iff $tv(\mathcal{M}(x), \mathcal{M}(x')) \leq \epsilon$ for all $x, x' \in \mathcal{X}$.

For systems modelled by FPAs, by (6.4) and Theorem 6.3.10, we have that if $bm(s_x, s_{x'}) \leq \epsilon$ for all $x, x' \in \mathcal{X}$, then $\mathcal{M}$ satisfies ϵ-weak probabilistic anonymity. Hence bm can be used to verify this anonymity property.

6.5 Non-expansiveness

Process algebras provide the link to the desired compositional reasoning about approximate equality in such a pseudometric framework. In order to specify and verify systems in a compositional manner, it is necessary that the behavioral semantics is compatible with all operators of the language that describe these systems. For behavioral equivalence semantics there is the common agreement that compositional reasoning requires that the considered behavioral equivalence is a congruence wrt all operators. On the other hand, for behavioral metric semantics there are several proposals of properties that operators should satisfy in order to facilitate compositional reasoning [DJGP04, BBLM13a].

Non-expansiveness is the most widely studied compositionality property. In this section we select three most typical non-recursive constructs of process algebra: nondeterministic choice, probabilistic choice and parallel composition, showing that they are non-expansive w.r.t. bisimilarity metrics. More precisely, we find a universal bound for the nondeterministic choice operator w.r.t. the generalized bisimilarity metric bm_V. We give the according bounds for the other two constructs w.r.t. $bm_\otimes$. In fact, we will provide upper bounds on the distance between the composed processes which are in case of the nondeterministic and probabilistic composition even stricter than the non-expansiveness condition.

Non-expansiveness corresponds to the quantitative analogue of congruence in behavioral equivalence, stating that the distance between composed

processes is at most the sum of the distance between its parts.

Definition 6.5.1 (Non-expansiveness)**.** *An n-ary operator f is* non-expansive *wrt a pseudometric m if*

$$m(f(s_1, \ldots, s_n), f(t_1, \ldots, t_n)) \leq \sum_{i=1}^{n} m(s_i, t_i)$$

When f is a binary operator, which is common for process algebra, by the property of the triangle inequality of m, to prove the non-expansiveness of f, it suffices to show that f satisfies the following condition:

Proposition 6.5.2. *A binary operator f is non-expansive wrt a pseudometric m if*

$$m(f(s,t), f(s,t')) \leq m(t,t') \text{ and } m(f(s,t), f(s',t)) \leq m(s,s')$$

Proof. By assumption, we have $m(f(s,t), f(s,t')) \leq m(t,t')$ and $m(f(s,t'), f(s',t')) \leq m(s,s')$. By the triangle inequality of m, we get $m(f(s,t), f(s',t'))$ $\leq m(f(s,t), f(s,t')) + m(f(s,t'), f(s',t')) \leq m(t,t') + m(s,s')$ as required. □

We first show a general bound for the nondeterministic choice operator w.r.t. bm_V. The importance of this result lies in that this bound holds for all the metrics in the family of generalized bisimilarity metrics, rather than just for a particular member.

Theorem 6.5.3. *Let s, t, s', t' be probabilistic processes. Then*

$$bm_V(s+t, s'+t') \leq \max(bm_V(s,s'), bm_V(t,t'))$$

Proof. We sketch the proof as follows. By Def. 6.3.1 we get that

$$F_V(bm_V)(s+t, s'+t') \leq \max\{F_V(bm_V)(s,s'), F_V(bm_V)(t,t')\}$$

Using the fact $bm_V = F_V(bm_V)$ completes the proof of the required result. □

Now we move to the study of non-expansiveness w.r.t. the multiplicative one $bm_\otimes$. We start by showing an important auxiliary property how the distance between convex combinations of probability distributions relates to the distance between the combined probability distributions.

Proposition 6.5.4. *Let $\mu_1, \mu_2, \mu_1', \mu_2' \in Disc(S)$ and $p \in [0,1]$. Then*

$$K_\otimes(bm_\otimes)(p\mu_1+(1-p)\mu_2, p\mu_1'+(1-p)\mu_2') \leq \max(K_\otimes(bm_\otimes)(\mu_1,\mu_1'), K_\otimes(bm_\otimes)(\mu_2,\mu_2'))$$

Proof. By the definition of $K_\otimes(bm_\otimes)$,

$$K_\otimes(bm_\otimes)(p\mu_1+(1-p)\mu_2, p\mu_1'+(1-p)\mu_2') = \max |\ln \frac{\sum_i [p\mu_1(s_i)+(1-p)\mu_2(s_i)]x_i}{\sum_i [p\mu_1'(s_i)+(1-p)\mu_2'(s_i)]x_i}|$$

under the constraints: $\forall i,j.\ x_i \leq e^{bm_\otimes(s_i,s_j)} x_j$. Let x_i^*'s be the variables that realize the maximum value on the problem. We have:

$$\begin{aligned}
& K_\otimes(bm_\otimes)(p\mu_1 + (1-p)\mu_2, p\mu_1' + (1-p)\mu_2') \\
= \ & \ln \frac{\sum_i [p\mu_1(s_i)+(1-p)\mu_2(s_i)]x_i^*}{\sum_i [p\mu_1'(s_i)+(1-p)\mu_2'(s_i)]x_i^*} \\
= \ & \ln \frac{p\sum_i \mu_1(s_i)x_i^* + (1-p)\sum_i \mu_2(s_i)x_i^*}{p\sum_i \mu_1'(s_i)x_i^* + (1-p)\sum_i \mu_2'(s_i)x_i^*} \\
\leq \ & \ln \frac{e^{K_\otimes(bm_\otimes)(\mu_1,\mu_1')} p\sum_i \mu_1'(s_i)x_i^* + e^{K_\otimes(bm_\otimes)(\mu_2,\mu_2')}(1-p)\sum_i \mu_2'(s_i)x_i^*}{p\sum_i \mu_1'(s_i)x_i^* + (1-p)\sum_i \mu_2'(s_i)x_i^*} \\
\leq \ & \ln \frac{e^{\max\{K_\otimes(bm_\otimes)(\mu_1,\mu_1'), K_\otimes(bm_\otimes)(\mu_2,\mu_2')\}} \left(p\sum_i \mu_1'(s_i)x_i^* + (1-p)\sum_i \mu_2'(s_i)x_i^*\right)}{p\sum_i \mu_1'(s_i)x_i^* + (1-p)\sum_i \mu_2'(s_i)x_i^*} \\
\leq \ & \max\{K_\otimes(bm_\otimes)(\mu_1,\mu_1'), K_\otimes(bm_\otimes)(\mu_2,\mu_2')\}
\end{aligned}$$

in which the first inequality is obtained by the definition of $K_\otimes(bm_\otimes)$:

$$\ln \frac{\sum_i \mu_1(s_i)x_i^*}{\sum_i \mu_1'(s_i)x_i^*} \leq K_\otimes(bm_\otimes)(\mu_1,\mu_1')$$

and

$$\ln \frac{\sum_i \mu_2(s_i)x_i^*}{\sum_i \mu_2'(s_i)x_i^*} \leq K_\otimes(bm_\otimes)(\mu_2,\mu_2')$$

□

Now we can show that the following operators are non-expansive w.r.t. $bm_\otimes$.

Theorem 6.5.5. *Let s, t, s', t' be probabilistic processes. Then*

1. $bm_\otimes(s \oplus_p t, s' \oplus_p t') \leq \max(bm_\otimes(s, s'), bm_\otimes(t, t'))$
2. $bm_\otimes(s \,|\, t, s' \,|\, t') \leq bm_\otimes(s, s') + bm_\otimes(t, t')$

Proof. Case *1*:

$$
\begin{aligned}
& bm_\otimes(s \oplus_p t, s' \oplus_p t') \\
=& F_\otimes(bm_\otimes)(s \oplus_p t, s' \oplus_p t') \quad \text{Def. 6.3.4} \\
=& \max_{a \in A} \Big\{ \sup_{s \xrightarrow{a} \mu_1, t \xrightarrow{a} \mu_2} \inf_{s' \xrightarrow{a} \mu_1', t' \xrightarrow{a} \mu_2'} K_\otimes(bm_\otimes)(p\mu_1 + (1-p)\mu_2, p\mu_1' + (1-p)\mu_2'), \\
& \quad \sup_{s' \xrightarrow{a} \mu_1', t' \xrightarrow{a} \mu_2'} \inf_{s \xrightarrow{a} \mu_1, t \xrightarrow{a} \mu_2} K_\otimes(bm_\otimes)(p\mu_1' + (1-p)\mu_2', p\mu_1 + (1-p)\mu_2) \Big\} \quad \text{Def. 6.3.1} \\
\leq& \max_{a \in A} \Big\{ \sup_{s \xrightarrow{a} \mu_1, t \xrightarrow{a} \mu_2} \inf_{s' \xrightarrow{a} \mu_1', t' \xrightarrow{a} \mu_2'} \max(K_\otimes(bm_\otimes)(\mu_1, \mu_1'), K_\otimes(bm_\otimes)(\mu_2, \mu_2')), \\
& \quad \sup_{s' \xrightarrow{a} \mu_1', t' \xrightarrow{a} \mu_2'} \inf_{s \xrightarrow{a} \mu_1, t \xrightarrow{a} \mu_2} \max(K_\otimes(bm_\otimes)(\mu_2, \mu_2'), K_\otimes(bm_\otimes)(\mu_1, \mu_1')) \Big\} \quad \text{Prop. 6.5.4} \\
\leq& \max_{a \in A} \max \Big\{ \sup_{s \xrightarrow{a} \mu_1} \inf_{s' \xrightarrow{a} \mu_1'} K_\otimes(bm_\otimes)(\mu_1, \mu_1'), \sup_{t \xrightarrow{a} \mu_2} \inf_{t' \xrightarrow{a} \mu_2'} K_\otimes(bm_\otimes)(\mu_2, \mu_2'), \\
& \quad \sup_{t' \xrightarrow{a} \mu_2'} \inf_{t \xrightarrow{a} \mu_2} K_\otimes(bm_\otimes)(\mu_2, \mu_2'), \sup_{s' \xrightarrow{a} \mu_1'} \inf_{s \xrightarrow{a} \mu_1} K_\otimes(bm_\otimes)(\mu_1, \mu_1') \Big\} \\
\leq& \max \Big(\max_{a \in A} \{ \sup_{s \xrightarrow{a} \mu_1} \inf_{s' \xrightarrow{a} \mu_1'} K_\otimes(bm_\otimes)(\mu_1, \mu_1'), \sup_{s' \xrightarrow{a} \mu_1'} \inf_{s \xrightarrow{a} \mu_1} K_\otimes(bm_\otimes)(\mu_1, \mu_1') \}, \\
& \quad \max_{a \in A} \{ \sup_{t' \xrightarrow{a} \mu_2'} \inf_{t \xrightarrow{a} \mu_2} K_\otimes(bm_\otimes)(\mu_2, \mu_2'), \sup_{t \xrightarrow{a} \mu_2} \inf_{t' \xrightarrow{a} \mu_2'} K_\otimes(bm_\otimes)(\mu_2, \mu_2') \} \Big) \\
=& \max(bm_\otimes(s, s'), bm_\otimes(t, t')) \quad \text{Def. 6.3.4 and 6.3.1}
\end{aligned}
$$

Case *2*: By Prop. 6.5.2, it suffices to show that

$$bm_\otimes(s \,|\, t, s \,|\, t') \leq bm_\otimes(t, t')$$

We construct a metric m as:

$$m(Q,R) = \begin{cases} bm_{\otimes}(t,t') & \text{if } Q = s \mid t \text{ and } R = s \mid t' \\ 0 & \text{if } Q = R \\ \infty & \text{otherwise.} \end{cases} \tag{6.9}$$

For any $\epsilon \geq 0$, if $m(s \mid t, s \mid t') \leq \epsilon$, then also $bm_{\otimes}(t,t') \leq \epsilon$.

We assume that if there exists one transition from t labelled by a, then there exists also one transition from t' labelled by a. Otherwise, $bm_{\otimes}(t,t') = \infty$ which completes the proof.

If $s \mid t \xrightarrow{a} \nu$ is due to one transition from t: $t \xrightarrow{a} \mu$, then by rule PAR1 (Fig. 2.1) $\nu = s \mid \mu$. If there exists on transition from t' s.t. $t' \xrightarrow{a} \mu'$ and $s \mid t' \xrightarrow{a} s \mid \mu'$. By the definition of $K_{\otimes}$ (Def. 6.4.1), we can check that $K_{\otimes}(m)(s \mid \mu, s \mid \mu') = K_{\otimes}(\mu,\mu') \leq \epsilon$.

If $s \mid t \xrightarrow{a} \nu$ is due to two transitions $s \xrightarrow{b} \delta(s_1)$ and $t \xrightarrow{\overline{b}} \delta(t_1)$, then a must be τ and by rule COM (Fig. 2.1) $\nu = \delta(s_1 \mid t_1)$. Also there exists one transition from t' s.t. $t' \xrightarrow{\overline{b}} \delta(t'_1)$ and $s \mid t' \xrightarrow{\tau} \delta(s_1 \mid t'_1)$. By the definition of $K_{\otimes}$, we deduce that $K_{\otimes}(m)(\delta(s_1 \mid t_1), \delta(s_1 \mid t'_1)) = m(s_1 \mid t_1, s_1 \mid t'_1) = bm_{\otimes}(t_1, t'_1) \leq \epsilon$.

The case when $s \mid t \xrightarrow{a} \nu$ is due to a transition from s is trivial.

By the definition that the function $F_{\otimes}$ is the Hausdorff distance between the transitions of $s \mid t$ and $s \mid t'$, we obtain $F_{\otimes}(m)(s \mid t, s \mid t') \leq \epsilon$. Thus, the constructed metric m satisfies $F_{\otimes}(m) \preceq m$, namely, it is a pre-fixpoint of $F_{\otimes}$. Remember that $bm_{\otimes}$ is the least one, thus we have $bm_{\otimes}(s \mid t, s \mid t') \leq m(s \mid t, s \mid t') = bm_{\otimes}(t,t')$ as required. □

A similar result can be gained for the bisimilarity metric bm based on the standard Kantorovich lifting. This generalizes a similar result of [DJGP04] which considered only probabilistic transition systems without nondeterministic branching.

6.6 Conclusion

We have proposed a family of Kantorovich pseudometrics depending on the notion of distance used to specify properties over traces. We have developed the theory of this notion, and showed how we can use it to verify the corresponding kind of properties. We have also showed that for the multiplicative variant, which is an interesting case because of its relation with differential privacy, it is possible to give a dual form that makes the metric computable by standard techniques.

Seven

Conclusion

In this thesis, we presented three ways of analyzing differentially private behaviors in concurrent setting. They are compositional, metrical and axiomatic methods, respectively. In Chapter 3 of the thesis, we focused on the compositional method and examined how the composition under each process constructor affects the level of differential privacy of the resulting system, which makes it possible to deduce the degree of differential privacy of a system from its sub-components. In Chapter 4, we borrowed the idea of amortisation and coined an amortised probabilistic bisimulation. It allows us to verify differential privacy, and it was shown to be a more liberal notion than the work in [TKD11]. In Chapter 5 we moved to the axiomatic method and provided sound and complete proof systems for our amortised bisimulation and its weak counterpart. Chapter 6 presents an extension of the bisimulation metric based on the Kantorovich distance. The standard notion is additive in nature and therefore not suitable to prove the property of differential privacy (which is multiplicative), the extension developed in the thesis is parametric with respect to the underlying distance, and therefore suitable to capture a vast range of properties, including differential privacy.

A great deal of future work may be worth considering on these issues. One interesting direction would be to explore the applicability of our modular reasoning approach to the problem of preserving privacy in geolocation-

related applications. More specifically, we intend to use (a possibly extended version of) our probabilistic process calculus to express systems of concurrent agents moving in space and time, and interacting with each other in ways that depend on the adjacency relation. We believe that our compositional method will provide a way to synthesize differentially private mechanisms in a (semi-)automatic way.

The pseudometric based on the Kantorovich metric proposed in [DJGP02] was recently axiomatized in [DGL14]. We have generalized this pseudometric to a family of pseudometrics in Chapter 6. It would be exciting to find an axiomatization for the family of generalized bisimulation metrics, thus obtaining a general framework of inference systems for probabilistically behavioral pseudometrics. As another future extension of the axiomatic theories developed in this thesis, we would like to extend the process language considered in the proof systems with recursion, following the lines of [DPP05, DP05].

For the computation of the pseudometric [DJGP02] based on the standard Kantorovich distance, thanks to its fixed point characterization, several iterative algorithms have been developed in order to compute its approximation up to any degree of accuracy [FPP04, vBW06, BSW07]. Recently, Chen *et. al.* [CvBW12] proved that, for finite fully probabilistic automata with rational transition function, the bisimilarity pseudometrics can be computed exactly in polynomial time. Despite its theoretical importance, the method used in the proof is known to be inefficient in practice. In [BBLM13b], an on-the-fly algorithm was proposed for computing the exact behavioral pseudometric in practice. One very important and natural future work is the investigation of methods to compute other members in the family of generalized bisimulation metrics and of conditions that make possible a general dual form. It would be interesting to see how the aforementioned algorithmic ideas can be exploited for the computation of the metrics in the family.

Bibliography

[ABLS14] Alessandro Aldini, Alessandro Bogliolo, Carlos Ballester Lafuente, and Jean-Marc Seigneur. On the tradeoff among trust, privacy, and cost in incentive-based networks. In Jianying Zhou, Nurit Gal-Oz, Jie Zhang, and Ehud Gudes, editors, *IFIPTM*, volume 430 of *IFIP Advances in Information and Communication Technology*, pages 205–212. Springer, 2014.

[AG99] Martín Abadi and Andrew D. Gordon. A calculus for cryptographic protocols: The spi calculus. *Inf. and Comp.*, 148(1):1–70, 1999.

[APSVR11] Miguel E. Andrés, Catuscia Palamidessi, Ana Sokolova, and Peter Van Rossum. Information Hiding in Probabilistic Concurrent Systems. *Journal of Theor. Comp. Sci.*, 412(28):3072–3089, 2011.

[APvRS10] Miguel E. Andrés, Catuscia Palamidessi, Peter van Rossum, and Geoffrey Smith. Computing the leakage of information-hiding systems. In *Proc. of TACAS*, volume 6015 of *LNCS*, pages 373–389. Springer, 2010.

[BBLM13a] Giorgio Bacci, Giovanni Bacci, Kim G Larsen, and Radu Mardare. Computing Behavioral Distances, Compositionally. In *Proc. MFCS'13*, pages 74–85. Springer, 2013.

[BBLM13b] Giorgio Bacci, Giovanni Bacci, Kim G. Larsen, and Radu Mardare. On-the-fly exact computation of bisimilarity distances. In *TACAS*, volume 7795 of *LNCS*, pages 1–15. Springer, 2013.

[BCP08] Christelle Braun, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. Compositional methods for information-hiding. In *Proc. of FOSSACS*, volume 4962 of *LNCS*, pages 443–457. Springer, 2008.

[BDG+13] Gilles Barthe, George Danezis, Benjamin Grégoire, César Kunz, and Santiago Zanella Béguelin. Verified computational differential privacy with applications to smart metering. In *CSF*, pages 287–301, 2013.

[BK84] J.A. Bergstra and J.W. Klop. Process algebra for synchronous communication. *Information and Control*, 60(1,3):109–137, 1984.

[BKOB12] Gilles Barthe, Boris Köpf, Federico Olmedo, and Santiago Z. Béguelin. Probabilistic relational reasoning for differential privacy. In *Proc. of POPL*. ACM, 2012.

[Bor06] Michele Boreale. Quantifying information leakage in process calculi. In *Automata, Languages and Programming, 33rd Int. Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proc., Part II*, volume 4052 of *LNCS*, pages 119–131. Springer, 2006.

[BP05] Mohit Bhargava and Catuscia Palamidessi. Probabilistic anonymity. In *Proc. of CONCUR*, volume 3653 of *LNCS*, pages 171–185. Springer, 2005.

[BS01] Emanuele Bandini and Roberto Segala. Axiomatizations for probabilistic bisimulation. In *Proc. of ICALP*, volume 2076 of *LNCS*, pages 370–381. Springer, 2001.

[BSW07] Franck Breugel, Babita Sharma, and James Worrell. Approximating a behavioural pseudometric without discount for probabilistic systems. In Helmut Seidl, editor, *Foundations of Software Science and Computational Structures*, volume 4423 of *Lecture Notes in Computer Science*, pages 123–137. Springer Berlin Heidelberg, 2007.

[BW90] J.C.M. Baeten and W.P. Weijland. *Proces algebra*, volume 18 of *Cambridge tracts in theoretical computer science*. CUP, 1990.

[CABP13] Konstantinos Chatzikokolakis, Miguel E. Andrés, Nicolás Emilio Bordenabe, and Catuscia Palamidessi. Broadening the scope of differential privacy using metrics. In *Privacy Enhancing Technologies*, pages 82–102, 2013.

[CAD] CADP: Construction and Analysis of Distributed Processes - Software Tools for Designing Reliable Protocols and Systems. http://cadp.inria.fr/.

[CCK+06] Ran Canetti, Ling Cheung, Dilsun Kaynar, Moses Liskov, Nancy Lynch, Olivier Pereira, and Roberto Segala. Task-structured probabilistic i/o automata. In *Proc. of WODES*, 2006.

[CdAMR08] Krishnendu Chatterjee, Luca de Alfaro, Rupak Majumdar, and Vishwanath Raman. Algorithms for Game Metrics. In *FSTTCS*, volume 2, pages 107–118. Leibniz-Zentrum fuer Informatik, 2008.

[CG09] Xiaojuan Cai and Yonggen Gu. Measuring anonymity. In *ISPEC*, volume 5451 of *LNCS*, pages 183–194. Springer, 2009.

[CGPX14] Konstantinos Chatzikokolakis, Daniel Gebler, Catuscia Palamidessi, and Lili Xu. Generalized bisimulation met-

rics. In *Proc. of CONCUR*, volume 8704 of *LNCS*, pages 32–46. Springer, 2014.

[Cha88] David Chaum. The dining cryptographers problem: Unconditional sender and recipient untraceability. *Journal of Cryptology*, 1:65–75, 1988.

[Che08] Steve Cheng. A crash course on the lebesgue integral and measure theory, 2008.

[CHM05] David Clark, Sebastian Hunt, and Pasquale Malacaria. Quantitative information flow, relations and polymorphic types. *J. of Logic and Computation*, 18(2):181–199, 2005.

[CNP09] Konstantinon Chatzikokolakis, Gethin Norman, and David Parker. Bisimulation for demonic schedulers. In *Proc. of FOSSACS*, volume 5504 of *LNCS*, pages 318–332. Springer, 2009.

[CP06] Konstantinos Chatzikokolakis and Catuscia Palamidessi. Probable innocence revisited. *Theor. Comp. Sci.*, 367(1-2):123–138, 2006.

[CP10] Konstantinos Chatzikokolakis and Catuscia Palamidessi. Making random choices invisible to the scheduler. *Information and Computation*, 208(6):694–715, 2010.

[CP12] Gheorghe Comanici and Doina Precup. Basis function discovery using spectral clustering and bisimulation metrics. In *AAAI*, volume 7113 of *LNCS*, pages 85–99. Springer, 2012.

[CPB] Konstantinos Chatzikokolakis, Catuscia Palamidessi, and Christelle Braun. Compositional methods for information-hiding. To appear in *Mathematical Structures in Computer Science*.

[CSWH00] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore W. Hong. Freenet: A distributed anonymous information storage and retrieval system. In *Proc. of DIAU*, volume 2009 of *LNCS*, pages 44–66. Springer, 2000.

[CT91] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. J. Wiley & Sons, Inc., 1991.

[CvBW12] Di Chen, Franck van Breugel, and James Worrell. On the complexity of computing probabilistic bisimilarity. In *FOSSACS*, volume 7213 of *LNCS*, pages 437–451. Springer, 2012.

[DBMC14] Anupam Das, Nikita Borisov, Prateek Mittal, and Matthew Caesar. Re^3: relay reliability reputation for anonymity systems. In Shiho Moriai, Trent Jaeger, and Kouichi Sakurai, editors, *ASIACCS*, pages 63–74. ACM, 2014.

[DD09] Yuxin Deng and Wenjie Du. The Kantorovich metric in computer science: A brief survey. *Electr. Notes Theor. Comput. Sci.*, 253(3):73–82, 2009.

[dFERVGR07] David de Frutos-Escrig, Fernando Rosa-Velardo, and Carlos Gregorio-Rodríguez. New bisimulation semantics for distributed systems. In *FORTE*, pages 143–159, 2007.

[DGJP99] Josee Desharnais, Vineet Gupta, Radha Jagadeesan, and Prakash Panangaden. Metrics for labeled Markov systems. In *CONCUR*, volume 1664 of *Lecture Notes in Computer Science*, pages 258–273. Springer, 1999.

[DGL14] Pedro R. D'Argenio, Daniel Gebler, and Matias David Lee. Axiomatizing bisimulation equivalences and metrics from probabilistic SOS rules. In *Proc. of FoSSaCS*, volume 8412 of *LNCS*, pages 289–303. Springer, 2014.

[DH13] Yuxin Deng and Matthew Hennessy. Compositional reasoning for weighted Markov decision processes. *Sci. Comput. Program.*, 78(12):2537–2579, 2013.

[DJGP02] Josee Desharnais, Radha Jagadeesan, Vineet Gupta, and Prakash Panangaden. The metric analogue of weak bisimulation for probabilistic processes. In *Proc. of LICS*, pages 413–422. IEEE, 2002.

[DJGP04] Josee Desharnais, Radha Jagadeesan, Vineet Gupta, and Prakash Panangaden. Metrics for labelled Markov processes. *Theor. Comp. Sci.*, 318(3):323–354, 2004.

[DKM+06] Cynthia Dwork, Krishnaram Kenthapadi, Frank Mcsherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *In EUROCRYPT*, pages 486–503. Springer, 2006.

[DL09] Cynthia Dwork and Jing Lei. Differential privacy and robust statistics. In *Proc. of the 41st Annual ACM Symposium on Theory of Computing (STOC)*, pages 371–380. ACM, 2009.

[DLT08] Josée Desharnais, François Laviolette, and Mathieu Tracol. Approximate analysis of probabilistic processes: Logic, simulation and games. In *QEST*, pages 264–273. IEEE Computer Society, 2008.

[DMS04] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. In *Proc. of the 13th USENIX Security Symposium*, 2004.

[DP05] Yuxin Deng and Catuscia Palamidessi. Axiomatizations for probabilistic finite-state behaviors. In *Proc. of FOSSACS*, volume 3441 of *LNCS*, pages 110–124. Springer, 2005.

[DPP05] Yuxin Deng, Catuscia Palamidessi, and Jun Pang. Compositional reasoning for probabilistic finite-state behaviors. In

Processes, Terms and Cycles: Steps on the Road to Infinity, volume 3838 of *LNCS*, pages 309–337. Springer, 2005.

[DPP07] Yuxin Deng, Catuscia Palamidessi, and Jun Pang. Weak probabilistic anonymity. In *Proc. of the 3rd Int. Workshop on Security Issues in Concurrency (SecCo)*, volume 180 (1) of *ENTCS*, pages 55–76. Elsevier, 2007.

[DPW06] Yuxin Deng, Jun Pang, and Peng Wu. Measuring anonymity with relative entropy. In *Proc. of the 4th Int. Worshop on Formal Aspects in Security and Trust*, volume 4691 of *LNCS*, pages 65–79. Springer, 2006.

[Dwo06] Cynthia Dwork. Differential privacy. In *Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Proceedings, Part II*, volume 4052 of *LNCS*, pages 1–12. Springer, 2006.

[Dwo11] Cynthia Dwork. A firm foundation for private data analysis. *Communications of the ACM*, 54(1):86–96, 2011.

[FG00] Riccardo Focardi and Roberto Gorrieri. Classification of security properties (part i: Information flow). In *FOSAD*, pages 331–396, 2000.

[FPP04] Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite Markov decision processes. In *UAI*, pages 162–169. AUAI Press, 2004.

[FPP05] Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for Markov decision processes with infinite state spaces. In *UAI*, pages 201–208. AUAI Press, 2005.

[FPP11] Norm Ferns, Prakash Panangaden, and Doina Precup. Bisimulation metrics for continuous Markov decision processes. *SIAM J. Comput*, 40(6):1662–1714, 2011.

[GHH+13] Marco Gaboardi, Andreas Haeberlen, Justin Hsu, Arjun Narayan, and Benjamin C. Pierce. Linear dependent types for differential privacy. In *POPL*, pages 357–370, 2013.

[Hen11] Matthew Hennessy. A calculus for costed computations. *Logical Methods in Computer Science*, 7(1), 2011.

[HJ90] Hans Hansson and Bengt Jonsson. A calculus for communicating systems with time and probabitilies. In *Proceedings of the Real-Time Systems Symposium - 1990*, pages 278–287. IEEE Computer Society, 1990.

[HL93] Matthew Hennessy and Huimin Lin. Proof systems for message-passing process algebras. In *CONCUR*, volume 715 of *LNCS*, pages 202–216. Springer, 1993.

[HO05] Joseph Y. Halpern and Kevin R. O'Neill. Anonymity and information hiding in multiagent systems. *J. of Comp. Security*, 13(3):483–512, 2005.

[Hoa85] C. A. R. Hoare. *Communicating Sequential Processes*. Prentice-Hall, 1985.

[HPSE10] Sardaouna Hamadou, Catuscia Palamidessi, Vladimiro Sassone, and Ehab ElSalamouny. Probable innocence in the presence of independent knowledge. In *Postproceedings of the 6th Int. Workshop on Formal Aspects in Security and Trust*, volume 5983 of *LNCS*, pages 141–156. Springer, 2010.

[JJS13] Rob Jansen, Aaron Johnson, and Paul F. Syverson. Lira: Lightweight incentivized routing for anonymity. In *NDSS*. The Internet Society, 2013.

[JS90] Chi-Chang Jou and Scott A. Smolka. Equivalences, congruences, and complete axiomatizations for probabilistic processes. In *Proc. of CONCUR*, volume 458 of *LNCS*, pages 367–383. Springer, 1990.

[KAK05] Astrid Kiehn and S. Arun-Kumar. Amortised bisimulations. In *FORTE*, pages 320–334, 2005.

[Lah08] Sébastien Lahaie. How to take the dual of a linear program. Technical report, 2008. http://www.cs.columbia.edu/coms6998-3/lpprimer.pdf.

[Lin95] Huimin Lin. Pam: A process algebra manipulator. *Formal Methods in System Design*, 7(3):243–259, 1995.

[LS89] Kim G. Larsen and Arne Skou. Bisimulation through probabilistic testing. In *Proceedings of the 16th ACM Symposium on Principles of Programming Languages (POPL)*, pages 344–352, 1989.

[LS91] Kim G. Larsen and Arne Skou. Bisimulation through probabilistic testing. *Inf. and Comp.*, 94(1):1–28, 1991.

[LY00] Huimin Lin and Wang Yi. A proof system for timed automata. In *FoSSaCS*, volume 1784 of *LNCS*, pages 208–222. Springer, 2000.

[Mal07] Pasquale Malacaria. Assessing security threats of looping constructs. In *Proc. of POPL*, pages 225–235. ACM, 2007.

[McS09] Frank McSherry. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *Proc. of the ACM SIGMOD Int. Conf. on Management of Data*, pages 19–30. ACM, 2009.

[Mil89] Robin Milner. *Communication and Concurrency*. Series in Computer Science. Prentice Hall, 1989.

[MKA+08] Ashwin Machanavajjhala, Daniel Kifer, John M. Abowd, Johannes Gehrke, and Lars Vilhuber. Privacy: Theory meets practice on the map. In *Proc. of ICDE*, pages 277–286. IEEE, 2008.

[Mu09] Chunyan Mu. Measuring information flow in reactive processes. In *ICICS*, volume 5927 of *Lecture Notes in Computer Science*, pages 211–225. Springer, 2009.

[NDW10] Tsuen-Wan Ngan, Roger Dingledine, and Dan S. Wallach. Building incentives into tor. In Radu Sion, editor, *Financial Cryptography*, volume 6052 of *Lecture Notes in Computer Science*, pages 238–256. Springer, 2010.

[Nor91] Timothy Norfolk. When does a metric generate convex balls? Technical report, 1991. http://www.math.uakron.edu/~norfolk/convex.ps.

[NS09] Arvind Narayanan and Vitaly Shmatikov. De-anonymizing social networks. In *Proc. of S&P*, pages 173–187. IEEE, 2009.

[RP10] Jason Reed and Benjamin C. Pierce. Distance makes the types grow stronger: a calculus for differential privacy. In *Proceeding of the 15th ACM SIGPLAN international conference on Functional programming (ICFP)*, pages 157–168. ACM, 2010.

[RR98] Michael K. Reiter and Aviel D. Rubin. Crowds: anonymity for Web transactions. *ACM Trans. on Information and System Security*, 1(1):66–92, 1998.

[RS01] Peter Y. A. Ryan and Steve A. Schneider. Process algebra and non-interference. *Journal of Computer Security*, 9(1/2):75–103, 2001.

[RSK+10] Indrajit Roy, Srinath T. V. Setty, Ann Kilzer, Vitaly Shmatikov, and Emmett Witchel. Airavat: security and privacy for MapReduce. In *Proc. of the 7th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, pages 297–312. USENIX Association, 2010.

[Seg95] Roberto Segala. *Modeling and Verification of Randomized Distributed Real-Time Systems*. PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, June 1995. Available as Technical Report MIT/LCS/TR-676.

[SEH10] Vladimiro Sassone, Ehab ElSalamouny, and Sardaouna Hamadou. Trust in crowds: Probabilistic behaviour in anonymity protocols. In *Proc. of the Fifth Int. Symposium on Trustworthly Global Computing*, volume 6084 of *LNCS*, pages 88–102. Springer, 2010.

[SGR97] Paul F. Syverson, David M. Goldschlag, and Michael G. Reed. Anonymous connections and onion routing. In *Proc. of S&P*, pages 44–54, 1997.

[SHY10] Vladimiro Sassone, Sardaouna Hamadou, and Mu Yang. Trust in anonymity networks. In Paul Gastin and François Laroussinie, editors, *CONCUR*, volume 6269 of *Lecture Notes in Computer Science*, pages 48–70. Springer, 2010.

[SL94] Roberto Segala and Nancy Lynch. Probabilistic simulations for probabilistic processes. In *Proceedings of CONCUR*, volume 836 of *LNCS*, pages 481–496. Springer, 1994.

[SL95] Roberto Segala and Nancy Lynch. Probabilistic simulations for probabilistic processes. *Nordic Journal of Computing*, 2(2):250–273, 1995.

[Smi03] Geoffrey Smith. Probabilistic noninterference through weak probabilistic bisimulation. In *CSFW*, pages 3–13, 2003.

[Smi08] Adam Smith. Efficient, differentially private point estimators. *arXiv preprint arXiv:0809.4794*, 2008.

[Smi09] Geoffrey Smith. On the foundations of quantitative information flow. In *Proc. of FOSSACS*, volume 5504 of *LNCS*, pages 288–302. Springer, 2009.

[SV04] Ana Sokolova and Erik P. de Vink. Probabilistic automata: system types, parallel composition and comparison. In *Validation of Stochastic Systems: A Guide to Current Research*, volume 2925 of *LNCS*, pages 1–43. Springer, 2004.

[Tar55] Alfred Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5(2):285—309, 1955.

[TDZ11] Mathieu Tracol, Josée Desharnais, and Abir Zhioua. Computing distances between probabilistic automata. In *QAPL*, volume 57 of *EPTCS*, pages 148–162, 2011.

[TK10] D. Thorsley and E. Klavins. Approximating stochastic biochemical processes with Wasserstein pseudometrics. *Systems Biology, IET*, 4(3):193–211, May 2010.

[TKD11] Michael C. Tschantz, Dilsun Kaynar, and Anupam Datta. Formal verification of differential privacy for interactive systems (extended abstract). *Electron. Notes Theor. Comput. Sci.*, 276:61–79, sep 2011.

[vBW01a] Franck van Breugel and James Worrell. An algorithm for quantitative verification of probabilistic transition systems. In *Proc. of CONCUR'01*, pages 336–350. Springer, 2001.

[vBW01b] Franck van Breugel and James Worrell. Towards quantitative verification of probabilistic transition systems. In *Proc. of ICALP*, volume 2076 of *LNCS*, pages 421–432. Springer, 2001.

[vBW05] Franck van Breugel and James Worrell. A behavioural pseudometric for probabilistic transition systems. *Theor. Comp. Sci.*, 331(1):115–142, 2005.

[vBW06] Franck van Breugel and James Worrell. Approximating and computing behavioural distances in probabilistic transition systems. *Theor. Comp. Sci.*, 360(1-3):373 – 385, 2006.

[vBW14] Franck van Breugel and James Worrell. The complexity of computing a bisimilarity pseudometric on probabilistic automata. In *Horizons of the Mind*, volume 8464 of *LNCS*, pages 191–213. Springer, 2014.

[vGSS95] Rob J. van Glabbeek, Scott A. Smolka, and Bernhard Steffen. Reactive, generative, and stratified models of probabilistic processes. *Information and Computation*, 121(1):59–80, 1995.

[XCL14] Lili Xu, Konstantinos Chatzikokolakis, and Huimin Lin. Metrics for differential privacy in concurrent systems. In *Proc. of FORTE*, volume 8461 of *LNCS*, pages 199–215. Springer, 2014.

[XHP14] Lili Xu, Sardaouna Hamadou, and Catuscia Palamidessi. Privacy-preserving process constructors, 2014. Technical report.

[XL14] Lili Xu and Huimin Lin. Complete proof systems for amortised probabilistic bisimulations. *Journal of Computer Science and Technology*, 2014. To appear.

[Xu12] Lili Xu. Modular reasoning about differential privacy in a probabilistic process calculus. In *TGC*, volume 8191 of *LNCS*, pages 198–212. Springer, 2012.

[YSH12] Mu Yang, Vladimiro Sassone, and Sardaouna Hamadou. A game-theoretic analysis of cooperation in anonymity net-

works. In Pierpaolo Degano and Joshua D. Guttman, editors, *POST*, volume 7215 of *Lecture Notes in Computer Science*, pages 269–289. Springer, 2012.

Index