Query-preserving watermarking of relational databases and XML documents

David Gross-Amblard
Laboratoire Cedric - Cnam
292, rue Saint Martin
75141 PARIS Cedex 3, France
dgram@cnam.fr

ABSTRACT

Watermarking allows robust and unobtrusive insertion of information in a digital document. Very recently, techniques have been proposed for watermarking relational databases or XML documents, where information insertion must preserve a specific measure on data (e.g. mean and variance of numerical attributes.)

In this paper we investigate the problem of watermarking databases or XML while preserving a *set of parametric queries in a specified language*, up to an acceptable distortion.

We first observe that unrestricted databases can not be watermarked while preserving trivial parametric queries. We then exhibit query languages and classes of structures that allow guaranteed watermarking capacity, namely 1) local query languages on structures with bounded degree Gaifman graph, and 2) monadic second-order queries on trees or tree-like structures. We relate these results to an important topic in computational learning theory, the VC-dimension. We finally consider incremental aspects of query-preserving watermarking.

Categories and Subject Descriptors

H.1 [**Information Systems**]: Models and Principles; F.1.3 [**Theory of Computation**]: Complexity Measures and Classes; F.4 [**Theory of computation**]: Mathematical Logic and Formal Languages

General Terms

Theory, Security, Algorithms

Keywords

Watermarking, watermarking scheme, VC-dimension

PODS 2003, June 9-12, 2003, San Diego, CA.

Introduction

A growing part of Internet content is dynamically generated from databases. Classical situations can be captured by the following 3-tier model: *data owners* invest time and efforts to elaborate large and detailed databases, and wish to sell them to multiple *data servers*. Data servers buy such data, and answer queries through e.g. a web interface to several *final users*. A simple example is given by air travel information web-sites: timetables of all flights are possessed by a data owner. Data servers propose a search engine on these flights, answering queries asked by final users such as "flights from Paris to Delhi on the 1st of July". Other common examples are lodging information systems, meteorological and financial data, etc.

But data owners are exposed to malicious servers, trying to sell illegal copies as their own. This problem is strengthened by the digital nature of these informations, since perfect copies of a document are easily produced and disseminated. Thus, an important tool for data owners is the ability to argue ownership of a database, once a suspect one has been discovered, and to track back to the original malicious server.

Indirect access. A data owner may not have a direct access to the suspect database, since malicious servers try to evade detection. To bypass this problem, a data owner can also act as a final user, i.e. ask queries to the suspect server. Being able to prove ownership based on these only informations is a strong protection against illegal uses.

Watermarking. Informally, *watermarking* hides pertinent information in a document, such as the owner or purchaser's identity. This information is used to identify the real owner of a suspect document, or the original server who has performed an unauthorized diffusion. A "syntactic", invisible modification of the original document, such as adding `<owner></owner>` markups in a XML file, is not an efficient way of hiding informations, since a simple rewriting of the document is sufficient to erase the original owner's identity. Information hiding have to be more sophisticated and occur in a "semantic way", i.e. must impact on the document's quality.

Adversarial and non-adversarial models. In a naive setting, a data server will redistribute an identical copy of its document along with the hidden information (i.e the owner's

identity.) This model is called *non-adversarial*. In the *adversarial* setting, a malicious server will perform distortions on the document, in order to erase any identification mark. An important problem is then to design *robust* watermarking procedures, that resist to reasonable alterations of the document. Fortunately, these alterations can not be too large, since the data server still wants to distribute valuable data.

Database watermarking. Very recently, algorithms were proposed for watermarking structured data like XML documents and relational databases [1, 20]. In [1], Agrawal and Kiernan clearly identify the need for watermarking techniques in databases, and consider several important, databases-specific aspects, like incremental updatability of the watermark. In their setting, information insertion is performed by flipping bits in numerical attributes.

Based on experimental results, they observe that the mean and variance of all numerical attributes is preserved by this operation, showing that their technique may be sufficient for several situations. But they give no guarantee for the distortion induced on queries that a server may perform.

Query-preserving watermarking. In this paper we focus on the watermarking of databases, in the general setting where data servers perform queries in a language $\mathcal{L}$. The data owner has a valuable database instance, and data servers apply for a copy of this database, providing queries $\psi_1, \ldots, \psi_k$ they will answer to final users. These queries are parametrized by final user inputs. The problem is then to construct a *query-preserving watermarking scheme* that respects the following conditions:

- the scheme maps owner's database instance to several watermarked versions, and induces a small distortion on the results of queries $\psi_1(\bar{a}), \ldots, \psi_k(\bar{a})$, for any user input $\bar{a}$.
- the scheme can prove ownership based on answers to queries $\psi_1, \ldots, \psi_k$ only (the owner acts as any final user to get these answers.)

This watermarking is driven by what is important to the final user: results of queries. Notice that data servers may answer other queries to users, but only distortion on $\psi_1, \ldots, \psi_k$ is guaranteed. Of course, one tries to hide a large number of information bits with a small distortion. In this perspective, Agrawal and Kiernan's work can be viewed as a watermarking that only preserves (the mean of) a projection query on each numerical attribute, *without parameters*.

As in [1], we distinguish between *parameter values*, which can not be modified, because they act as parameters in queries, and numerical data that can be distorted (this implies that these numerical values do not act as keys, or are not part of an integrity constraint.) We denote the latter by *weight values*. Our watermarking schemes will modify the weighted part of a database instance, while leaving the parameter part unchanged. Attention is focused on numerical data for the sake of simplicity, but other domains with a distance function (e.g. strings with a similarity measure, or a semantic distance) can be considered.

The fundamental difficulty in query-preserving watermarking is to determine which weights to modify in order to get a unobtrusive insertion. Khanna and Zane [10] already gave some positive answers to this problem. They obtained a query-preserving watermarking scheme for a specific *parametric* query: shortest path queries on weighted graphs. Their information insertion does not modify the length of *any* shortest path beyond an acceptable and *provable* distortion. Moreover their scheme can prove ownership on a document based on query answers of the suspect data server. There is no need to have a direct access to the suspect database. They also provide a general method for watermarking in an adversarial setting, where malicious servers try to erase the watermark. From the theoretical point of view, they observe that shortest path queries have very low computational complexity, and suspect that watermarking schemes for NP-hard search spaces are difficult to analyze.

Contribution: watermarking and learning theory. Our first main result shows that the difficulty of query-preserving watermarking is linked to the *informational complexity* of sets defined by queries, rather than their computational complexity. This is related to an important combinatorial parameter in computational learning theory, the *Vapnik-Chervonenkis* dimension of sets (or VC-dimension [2, 22].) A finite VC-dimension for a family of sets is equivalent to its learnability (in the PAC model [22].) Roughly speaking, our result states that if the VC-dimension is not bounded but is maximal, no watermarking scheme can be obtained.

Recently, Grohe and Turàn [7] showed that the VC-dimension of sets defined by first-order logic and monadic second-order logic is bounded on restricted classes of structures, and this characterization is, in some sense, optimal. These restrictions concern bounding the degree of the Gaifman graph of the structure, or bounding its *tree-width*, which measures its similarity with trees. This last restriction has also fruitful applications in both database theory and computational complexity (see e.g. [4].)

Our second main result shows that under the same restrictions, a watermarking scheme can be obtained. First, we construct a watermarking scheme for database instances with bounded degree Gaifman graph, while preserving any *local* query. Local languages contain particularly first-order logic, order-invariant queries [6], and relational $\texttt{AGGR}_{\mathbb{Q}}$ queries [12, 13], that expresses mostly plain `SQL` by adding grouping and aggregate functions to relational calculus [8, 13, 14].

Second, we provide a watermarking scheme for first-order and monadic second-order queries on trees or tree-like structures. Monadic second-order logic (MSO) is of a special interest, since it is commonly used to model *pattern queries* on labeled trees, i.e. used as a formal query language for XML documents (see e.g. [16].) XML deals actually with unranked trees, but several methods exist to encode them into binary trees (as in [15]), so we will restrict our attention to the binary case.

Finally, on structures with unbounded degree Gaifman graph, one can construct a first-order formula that defines sets with unbounded and maximal VC dimension. There also exists an MSO-formula yielding such sets on structures with unbounded tree-width. For both, no query-preserving watermarking scheme can be obtained. This gives a rather complete panorama of query-preserving watermarking.

For practical applications, database instances are likely to have a bounded degree Gaifman graph or a bounded tree-width. A data owner can measure these combinatorial informations and estimate the watermarking capacity of the instance with our results.

Organization. The paper is organized as follows: we first give basic definitions on query-preserving watermarking, and recall the standard notion of the VC-dimension. We then show on section 2 that computing the exact watermarking capacity is hard. We prove that one can not obtain a general watermarking scheme for unrestricted database instances even for trivial queries and relate this result to the VC-dimension. We then exhibit restrictions on database instances and query languages that allow watermarking with a reasonable amount of hidden information: local languages on structures with bounded degree Gaifman graph on section 3, monadic second-order queries on trees and tree-like structures on section 4. Finally, section 5 deals with incremental updatability of watermarked instances.

Related work. A wide part of the watermarking literature focuses on multimedia data including images, sound and video [3, 9]. Beside works cited in the introduction [1, 10], watermarking of structured data like trees, graphs, or solutions of an optimization problem are studied in [18, 19, 23]. A watermarking algorithm is also proposed for semi-structures like XML in [20], as part of the CERIAS project at Purdue University. These approaches do not consider the notion of queries.

1. BASIC DEFINITIONS

Weighted structures. A signature τ (or database schema) is a finite set of relation symbols $\{\mathbf{R_1}, \ldots, \mathbf{R_t}\}$, with respective arity $r_1, \ldots, r_t$. A finite structure $\mathcal{G} = \langle \mathcal{U}, R_1, \ldots, R_t \rangle$ (or database instance) is an interpretation of each relation symbol of the schema τ on a finite universe $\mathcal{U}$. We denote by $STRUCT[\tau]$ the set of all τ-structures. First-order formulas are built from atomic formulas on the database schema with equality, and are closed under classical boolean connectives $\wedge, \vee, \neg$ and quantifiers $\exists, \forall$. In monadic second-order logic (MSO), quantification is also on sets of elements. Given a formula $\psi(u_1, \ldots, u_r, v_1, \ldots, v_s)$, a structure $\mathcal{G}$ and $\bar{a} \in \mathcal{U}^r$, let $\psi(\bar{a}, \mathcal{G}) = \{\bar{b} \in \mathcal{U}^s : \mathcal{G} \models \psi(\bar{a}, \bar{b})\}$.

Similarly to [1], we suppose that elements from the finite universe $\mathcal{U}$ map to (i.e. are keys for) some numerical values, that our watermarking procedures will slightly modify in order to hide information. A *weighted structures* $(\mathcal{G}, \mathcal{W})$ is defined by a finite structure $\mathcal{G}$ and a weight assignment $\mathcal{W} : \mathcal{U}^s \to \mathbb{N}$ that maps a s-tuple $\bar{b}$ to its weight $\mathcal{W}(\bar{b})$ ($s \in \mathbb{N}$ is fixed by the schema.)

A formula with parameter $\bar{u}$ is a formula $\psi(\bar{u}, \bar{v})$ with two distinguished variable vectors, $\bar{u}$ and $\bar{v}$, such that $\bar{v}$ has arity s. Variables $\bar{u}$ can be assigned to a value $\bar{a}$ by a final user who wants to obtain the set $\mathcal{A}_{\bar{a}}^{(\mathcal{G},\mathcal{W}),\psi}$ of elements and weights corresponding to the query result:

$$\mathcal{A}_{\bar{a}}^{(\mathcal{G},\mathcal{W}),\psi} = \{(\bar{b}, \mathcal{W}(\bar{b})) : \bar{b} \in \psi(\bar{a}, \mathcal{G})\}.$$

The set $\mathcal{A}_{\bar{a}}^{(\mathcal{G},\mathcal{W}),\psi}$ will be denoted $\mathcal{A}_{\bar{a}}$ for short, when $(\mathcal{G}, \mathcal{W})$ and ψ are clear.

EXAMPLE 1. *We consider the following database instance, with query $\psi(u, v) \equiv Route(u, v)$ registered by a server, and its translation into a weighted instance.*

Route:

travel	*transport*
India discovery	*F21*
India discovery	*G12*
Nepal Trek	*F21*
Nepal Trek	*R5*
Nepal Trek	*F2*
TourNepal	*F2*
TourNepal	*T33*

Timetable:

transport	*departure*	*arrival*	*type*	*duration*
F21	*Paris*	*Delhi*	*plane*	*10:35*
G12	*Delhi*	*Nawalgarh*	*bus*	*6:20*
R5	*Delhi*	*Kathmandu*	*plane*	*6:15*
F2	*Kathmandu*	*Simikot*	*plane*	*3:30*
T33	*Kathmandu*	*Daman*	*jeep*	*2:50*
G13	*Kathmandu*	*Paris*	*plane*	*10:00*

The only weight attribute is "duration". The corresponding structure is $\mathcal{G} = \langle \mathcal{U}, Route, Timetable \rangle$, with, as an example of tuple $(TourNepal, F2) \in Route$, $(F21, Paris, Delhi, plane) \in Timetable$ and $\mathcal{W}(F21) = 10:35$ (expressed in hours and minutes.)
There are only three possible parameters for the query, with e.g. $\mathcal{A}_{India\ discovery}^{Timetable,\psi} = \{(F21, 10:35), (G12, 6:20)\}$.

Query-preserving watermarking. Without loss of generality, we focus on the preservation of a unique query ψ, but extension to several queries $\psi_1, \ldots, \psi_k$ is straightforward by simple projection techniques.

Definition 1. A (water)marking problem is a pair $(\mathcal{K}, \psi)$, where $\mathcal{K}$ is a class of weighted structures on τ, and ψ a parametric query.

A watermarking algorithm will introduce perturbations into the structure's weight function $\mathcal{W}$, and these perturbations must be restricted.

Let $W_{\bar{a}}^{\mathcal{G},\psi} = \psi(\bar{a}, \mathcal{G})$ be the set of weighted elements involved in the computation of ψ for parameter $\bar{a}$. *It is noteworthy that $W_{\bar{a}}^{\mathcal{G},\psi}$ does not depend on the weight function $\mathcal{W}$: we can perturb $\mathcal{W}$ without modifying $W_{\bar{a}}^{\mathcal{G},\psi}$.*

The *weight* $f_{(\mathcal{G},\mathcal{W})}(\bar{a}, \psi)$ *of* $W_{\bar{a}}^{\mathcal{G},\psi}$ is defined by the sum of weights of its tuples:

$$f_{(\mathcal{G},\mathcal{W})}(\bar{a}, \psi) = \sum_{\bar{b} \in W_{\bar{a}}^{\mathcal{G},\psi}} \mathcal{W}(\bar{b}).$$

We will often use notations $W_{\bar{a}}$ and $f(\bar{a})$ only when $\mathcal{G}$, $\mathcal{W}$ and ψ are clear from the context.

Function f will be used to control the overall distortion induced on query results $\mathcal{A}_{\bar{a}}$ (the sum function can be replaced by *mean*, *min* or *max* without modifying the positive results of this paper.)

EXAMPLE 2. *For the database instance in example 1:*

$$f(India\ discovery) = 16:55,$$
$$f(Nepal\ Trek) = 20:20,$$
$$f(TourNepal) = 6:20.$$

Given a constant $c \in \mathbb{N}$, a weighted structure $(\mathcal{G}, \mathcal{W}')$ is said to satisfy the *c-local distortion assumption* with respect to another structure $(\mathcal{G}, \mathcal{W})$ if and only if for all $\bar{w} \in \mathcal{U}^s$, $|\mathcal{W}(\bar{w}) - \mathcal{W}'(\bar{w})| \leq c$. Furthermore, given $d \in \mathbb{N}$, it satisfies the *d-global distortion assumption* if and only if, for all

$\bar{a} \in \mathcal{U}^r$, $|f_{(\mathcal{G},\mathcal{W})}(\bar{a},\psi) - f_{(\mathcal{G},\mathcal{W}')}(\bar{a},\psi)| \leq d$. A structure is a *c-local distortion* (resp. *d-global distortion*) of another structure if it satisfies the c-local (resp. d-global) distortion assumption.

EXAMPLE 3. *We consider the original instance given in example 1 and the same query ψ. Let $Timetable'$ and $Timetable''$ be two possible distortions of Timetable:*

$Timetable'$:

transport	*departure*	*arrival*	*type*	*duration*
F21	*Paris*	*Delhi*	*plane*	*10:45*
G12	*Delhi*	*Nawal.*	*bus*	*6:30*
R5	*Delhi*	*Kathm.*	*plane*	*6:25*
F2	*Kathm.*	*Simikot*	*plane*	*3:20*
T33	*Kathm.*	*Daman*	*jeep*	*3:00*
G13	*Kathm.*	*Paris*	*plane*	*10:00*

$Timetable''$:

transport	*departure*	*arrival*	*type*	*duration*
F21	*Paris*	*Delhi*	*plane*	*10:25*
G12	*Delhi*	*Nawal.*	*bus*	*6:30*
R5	*Delhi*	*Kathm.*	*plane*	*6:05*
F2	*Kathm.*	*Simikot*	*plane*	*3:40*
T33	*Kathm.*	*Daman*	*jeep*	*2:40*
G13	*Kathm.*	*Paris*	*plane*	*10:00*

We have $\mathcal{A}^{Timetable',\psi}_{India\ discovery} = \{(F21, 10:45), (G12, 6:30)\}$. $Timetable'$ respects the c-local distortion assumption for constant $c = 0:10$, but not the d-global distortion assumption with respect to ψ for $d = 0:10$ (because $f_{Timetable'}(India\ discovery) = 17:15$.) $Timetable''$ respects both assumptions for $c = 0:10$ and $d = 0:10$.

If we can find 2^l distinct distortions of a database instance, we can distribute a distinct version to 2^l data servers, and hence identify 2^l possible malicious servers. Similarly, this means that we can hide l bits of information in the database instance. Each binary word will then constitute a different *mark*.

In the sequel we focus on algorithms producing structures that respect the local distortion assumption for a constant value, say $c = 1$ (i.e. weights are only modified by a $+1$ or -1 distortion.)

Active weighted elements.

Let $W^{\mathcal{G},\psi}$ be the *active weighted elements of* $(\mathcal{G},\mathcal{W})$ with respect to ψ, i.e.:

$$W^{\mathcal{G},\psi} = \bigcup_{\bar{a}\in\mathcal{U}^r} W^{\mathcal{G},\psi}_{\bar{a}}.$$

We will use notation W for short. In our example 1, active weighted elements are $\{\mathrm{F21}, \mathrm{G12}, \mathrm{R5}, \mathrm{F2}, \mathrm{T33}\}$, and $G13$ is inactive.

In the sequel, *we will only distort weights of active weighted elements*. As a consequence, there will be at most $|W|$ useful weights to modify. Distortions in example 3 respect this assumption. The next subsection will show why this restriction is used.

Watermarking procedures.

We now give some definitions in the spirit of [10] for watermarking structured data, that use probabilistic algorithms. A probabilistic algorithm has the ability to pick a random bit b at each step, and to adapt its computation according to the value of b. Hence a given computation is a path in the tree of all possible random choices along with its corresponding probability: both form a probability space Ω. It is convenient to consider such algorithms that may succeed with high probability and may fail, i.e. stop and abandon, or produce an incorrect result with a small probability δ.

Let $\mathcal{A}^{(\mathcal{G},\mathcal{W}),\psi} = \{\mathcal{A}^{(\mathcal{G},\mathcal{W}),\psi}_{\bar{a}} : \bar{a} \in \mathcal{U}^r\}$ be the set of all possible query answers from a server using $(\mathcal{G},\mathcal{W})$.

Definition 2. Given a formula ψ, and $l, d, d' \in \mathbb{N}$, $0 \leq \delta < 1$, a (l, d, d', δ)-marking procedure preserving ψ is a pair of probabilistic algorithms $\mathcal{M}$ and $\mathcal{D}$ such that:

1. $\mathcal{M}$ takes as an input an original structure $(\mathcal{G},\mathcal{W})$ and a boolean mark $m \in \{0,1\}^l$ and outputs a 1-local distortion $\mathcal{G}_m = (\mathcal{G},\mathcal{W}_m)$ such that:

   $$\Pr_{\Omega}[\mathcal{G}_m \text{respects the } d\text{-global distortion assumption}] > \frac{3}{4}.$$

2. Let $\mathcal{G}^* = (\mathcal{G},\mathcal{W}^*)$ be a d'-global distortion of $\mathcal{G}_m$. Algorithm $\mathcal{D}$ is such that, given as input structure $(\mathcal{G},\mathcal{W})$ and all possible answers $\mathcal{A}^{\mathcal{G}^*,\psi}$ from a suspect data server that uses $\mathcal{G}^*$:

   $$\Pr_{\Omega}[\mathcal{D} \text{ outputs } m] \geq 1 - \delta.$$

Algorithms $\mathcal{M}$ and $\mathcal{D}$ stand for the "marker" and the "detector", respectively. Parameter l stands for the number of bits to be hidden. Value d is the maximum acceptable global distortion on structures produced by the marker, and d' is the maximum global distortion an attacker can perform on a structure in order to erase the watermark. Finally, δ is the failure probability of the detector.

Marker $\mathcal{M}$ takes the binary message m to be hidden in the data, and computes the watermarked version of the original structure. The same marker is used for any of the 2^l different messages. Detector $\mathcal{D}$ identifies a suspect structure $\mathcal{G}^*$ based on query answers $\mathcal{A}^{\mathcal{G}^*,\psi}$ from the server.

Definition 3. A marking problem $(\mathcal{K},\psi)$ is said to have a marking procedure if there exists $0 \leq \delta < 1, l, d, d' \in \mathbb{N}$ and a pair $(\mathcal{M},\mathcal{D})$ that is a (l, d, d', δ)-marking procedures for structures in $\mathcal{K}$.

We recall that distortions are made on active weighted elements only. This leads to the two following observations. First, for 1-local distortions considered here, each weight can be modified by three means, i.e. a $+1$ or a -1 distortion, or no distortion at all. Hence, we consider at most $3^{|W|}$ different possible 1-local distortions of a structure, and the maximum number of bits one can encode is at most $O(|W|)$.

Second, these active weights can always be recovered from a suspect server by asking $\mathcal{A}_{\bar{a}}$ for all possible values of $\bar{a}$. But it is worth noting that modifying a weight $\mathcal{W}(\bar{a})$ of an element $\bar{a}$ outside W does not impact on servers answers $\mathcal{A}_{\bar{a}}$. This is *not* an efficient way to hide information, since *those weights will not be recoverable by querying the server*. Hence information insertion should arise from distortions in W only, and distortions outside W are useless.

Adversarial and non-adversarial model.

Constructing a correct global distortion $\mathcal{G}_m$ preserving ψ is a combinatorial problem on its own. The probabilistic aspect of the marker is useful, since we are going to produce correct structures with the probabilistic method, but a deterministic version can also be obtained. Once such a structure is produced, the following problem is to resist to attacks.

In the *non-adversarial model*, data servers do not modify the structure $\mathcal{G}_m$ they have received. Suspect structure $\mathcal{G}^*$ is exactly the watermarked structure $\mathcal{G}_m$, and answers $\mathcal{A}^{\mathcal{G}^*,\psi}$ are identical to $\mathcal{A}^{\mathcal{G}_m,\psi}$. So if there is a marker satisfying property 1 in definition 2, there is a detector satisfying property 2 with $\delta = 0$.

In the *adversarial model*, data servers can perform any reasonable distortion on the watermarked structure $\mathcal{G}_m$. In this case, a failure probability $\delta > 0$ is required for the detector. As a matter of fact, a natural attack is to guess the inserted mark and its position, and to modify the structure accordingly. Hopefully, the probability of this event will be small.

Watermarking schemes. A marking problem may have a marking procedure for a constant value of l. The interesting situation is when l is an increasing function of $|W|$, i.e. the number of hidden bits grows with the number of active weighted elements of the problem. The best situation would be to hide $|W|$ bits of data, without distorting results of queries at all, in such a way that the hidden bits can always be recovered. But there is a natural trade-off between $|W|$ and the global distortion.

Definition 4. A watermarking problem possesses a *marking scheme* if there exists $q \in \mathbb{N}$ such that the same pair of algorithms $(\mathcal{M}, \mathcal{D})$ with $0 < \varepsilon \leq \frac{1}{q}$ as parameter is a $(|W|^{1-q\varepsilon}, \frac{1}{\varepsilon}, d', \delta)$-marking procedure.

Naturally, the number of hidden bits increases with the allowed distortion (when $\varepsilon \to 0$.) For example, a scheme with $q = 1$ can hide $\sqrt{|W|}$ bits with distortion $\frac{1}{\varepsilon} = 2$ (this would be a very efficient scheme.)

Watermarking in the adversarial model. In this paper we restrict our attention to non-adversarial watermarking schemes only, but this is not a limitation. Indeed, Khanna and Zane [10] proposed a general technique to turn a non-adversarial scheme into an adversarial one. We recall this result here for completeness, and refer the reader to the original paper for a more precise exposition.

Two natural hypothesis are used to constraint the behavior of the attacker:

Assumption 1. Bounded distortion: the attacker respects the global distortion assumption, for an absolute constant d'.

Assumption 2. Limited knowledge: the attacker has limited knowledge on the mark distribution of the owner (the probability that an attacker constructs a weight function γ-close to the original, secret one is bounded by β, $0 < \beta, \gamma < 1$.)

The first assumption indicates that there is a limit to the distortion one can add to a structure, imposed by its intended use. The second simply says that the attacker does not know exactly what information has been introduced into the structure (and does not know the original, non-marked structure.) This models also the situation where a server is indeed not malicious, but uses data from an other source, similar to the owner's database (false positive detection.)

FACT 1. *[10] Under the Bounded distortion and Limited knowledge assumptions, any non-adversarial watermarking scheme can be turned into an adversarial one, with a constant error probability* $\max(\beta, o(1))$.

Observe that the watermarking robustness is obtained by lack of knowledge (an attacker knows there is a mark, but do not know its amplitude and distribution) and not by using the intractability of a computational problem, like in the cryptographic setting.

All the watermarking schemes presented in this paper comply with Khanna and Zane's framework, and support the adversarial and non-adversarial setting.

We do not consider here the general problem of *collusion attacks*, where servers combine several watermarked copies of the database to erase the watermark. Nevertheless, a specific notion of collusion is considered in section 5.

Vapnik-Chervonenkis dimension. Let V be a set and $\mathcal{C}$ be a family of subsets of V. A set $U \subseteq V$ is *shattered* by $\mathcal{C}$ if $\mathcal{C} \cap U = 2^U$, where $\mathcal{C} \cap U = \{C \cap U : C \in \mathcal{C}\}$. The VC-dimension $VC(\mathcal{C})$ of $\mathcal{C}$ with respect to V is the maximum of the sizes of the shattered subsets of V, or ∞ if the maximum does not exist. For a formula $\psi(\bar{u}, \bar{v})$ and a structure $\mathcal{G}$, let $\mathcal{C}(\psi, \mathcal{G}) = \{\psi(\bar{a}, \mathcal{G}) : \bar{a} \in \mathcal{U}^r\}$, and $VC(\psi, \mathcal{G}) = VC(\mathcal{C}(\psi, \mathcal{G}))$. We say that ψ has *bounded VC-dimension* on a class of structures $\mathcal{K}$ if there exists $k \in \mathbb{N}$ such that, for all $\mathcal{G} \in \mathcal{K}$, $VC(\psi, \mathcal{G}) \leq k$.

2. QUERY-PRESERVING WATERMARKING: GENERAL CASE

Computing the watermarking capacity. Computing the exact watermarking capacity $\#Mark$ of a class of structures, i.e. the number of different possible perturbations with distortion at most d is probably difficult. It appears that computing $\#Mark$ for distortion *exactly* d is as hard as computing the number of accepting paths of any NP Turing machine, i.e. is complete for the classical complexity class $\#P$ [21].

THEOREM 1. *$\#Mark(= d)$ is $\#P$-complete.*

PROOF. The problem $\#Mark(= d)$ is in $\#P$ by considering the NP-machine that guesses perturbations and checks the global d-distortion condition.

We show that $\#Mark(= d)$ is $\#P$-hard by reduction of the classical $\#P$-hard problem *PERMANENT* (i.e. counting the number of perfect matchings in a bipartite graph.)

Let $G = (V_1, V_2, E)$ be a bipartite graph. Let $\mathcal{U} = V_1 \cup V_2$ and $\mathcal{G}$ such that $\forall a \in \mathcal{U}$, $W_a = \{(u, v) : E(u, v)\}$. Constructing a weighted structure and a function ψ with such (W_a) is easy. Suppose now that we can compute $\#Mark((W_u), = d)$ for $d = 1$.

For all $b \in W$, let $\mathcal{W}'(b) = \mathcal{W}(b) + m_b$ be a possible watermarked weight function. It respects the following conditions:

$$\forall a \in \mathcal{U}, \sum_{b \in W_a} \mathcal{W}(b) + m_b - \mathcal{W}(b) = 1,$$

where m_b are under constraints

$$\forall b \in W, 0 \leq m_b \leq 1. \quad (+1\text{-weight distortion})$$

The number of possible values for m_b is exactly the number of perfect matchings of the previous graph G. □

Impossibility results. Guaranteed watermarking for arbitrary structures, preserving even trivial queries is impossible.

THEOREM 2. *A problem $(\mathcal{K}, \psi)$ does not possess a watermarking scheme if $\forall \mathcal{G} \in \mathcal{K}, VC(\psi, \mathcal{G}) = |W^{\mathcal{G},\psi}|$.*

PROOF. With at most k distorted weights, one can produce at most $\sum_{i=1}^{k} \binom{|W|}{i} 2^i \leq (2|W|)^k$ different weighted structures, encoding at most $O(k \ln |W|)$ bits. So any algorithm encoding $|W|^{1-q\varepsilon}$ bits must use a mark M with at least $h(|W|, \varepsilon) = \frac{|W|^{1-q\varepsilon}}{2 \ln |W|}$ distortions with the same sign, say $+1$. For a given ε_0, h is increasing with respect to $|W|$ (since $|W|^{1-q\varepsilon_0} > \ln |W|$), and there exists n_0 such that $h(n_0, \varepsilon_0) > \frac{1}{\varepsilon_0}$.

We now consider a structure $\mathcal{G}_{n_0}$ which universe has n_0 weighted elements. A watermarking scheme with parameter $\varepsilon = \varepsilon_0$ must add distortion $+1$ to weights from a set of elements P with $|P| > \frac{1}{\varepsilon_0}$.

Since $VC(\psi, \mathcal{G}_{n_0}) = |W^{\mathcal{G}_{n_0},\psi}|$, there exists a subset S of $\mathcal{U}^s$ of size $|W|$ which is shattered by sets in $\mathcal{C}(\psi, \mathcal{G}_{n_0})$. But since sets in $\mathcal{C}(\psi, \mathcal{G}_{n_0})$ are all subsets of W, there exists only one possible S: $S = W$ (sets in $\mathcal{C}(\psi, \mathcal{G}_{n_0})$ can not shatter sets outside W.) So the set W is shattered by results of queries, and there exists a tuple $\bar{a}$ such that $P = W_{\bar{a}}$. Hence distortion on $\psi(\bar{a}, \mathcal{G}_{n_0})$ is greater than $\frac{1}{\varepsilon_0}$, which contradicts the hypothesis to have a watermarking scheme. Probability and adversarial arguments does not come into play. □

It is worth noting that this impossibility argument can be followed with even trivial queries, e.g. $\psi(u, v) \equiv E(u, v)$. To do this, it is sufficient to consider the class of structures $\mathcal{G}_n$ with $2^n + n$ vertices, and the simple binary relation E that links the ith vertex of the first 2^n vertices to the ith subset W_i of the n last vertices.

REMARK 1. *Unbounded VC-dimension is not sufficient: one can construct a class of structures $\mathcal{G}_n$ of size n where only half of the active weights are shattered ($VC(\psi, \mathcal{G}_n) = |W|/2$), with a $(|W|/4, 0, \delta)$-marking scheme.*

Consider the class of structures $\mathcal{G}_n$ with $2^{n/2} + 1 + n$ vertices, and the simple binary relation E that links the ith vertex of the first $2^{n/2}$ vertices to the ith subset of the $n/2$ last vertices, and the $2^{n/2} + 1$th vertex a to all of the n last vertices. The watermarking problem defined by the query $\psi(u, v) = E(u, v)$ has n active weights and unbounded VC-dimension. The last $n/2$ vertices of the active weights are involved only for query $E(a, \mathcal{G}_n)$. Putting balanced distortions $(+1, -1)$ or $(-1, +1)$ only on these $n/2$ weights gives a watermarking scheme encoding $n/4$ bits with distortion 0.

3. WATERMARKING WHILE PRESERVING LOCAL QUERIES

Locality of queries. Given a structure $\mathcal{G} = \langle \mathcal{U}, R_1, \ldots, R_t \rangle$, its *Gaifman graph* is the new structure $\langle \mathcal{U}, E \rangle$, where $(a, b) \in E$ iff there is a relation R_i in $\mathcal{G}$ and a tuple $\bar{c}$ in R_i such that a and b appear in $\bar{c}$. The distance $d(a, b)$ between two elements a and b is the length of a shortest path between a and b in the Gaifman graph of $\mathcal{G}$. If no such path exists, $d(a, b) = \infty$. Given $a \in \mathcal{U}$, $\rho \in \mathbb{N}$, the ρ-sphere $S_\rho(a)$ is the set $\{b : d(a, b) \leq \rho\}$, and for a tuple $\bar{c}$, $S_\rho(\bar{c}) = \cup_{a \in \bar{c}} S_\rho(a)$. Given a tuple $\bar{c} = (c_1, \ldots, c_n)$, its ρ-neighborhood $N_\rho(\bar{c})$ is defined as the structure $\langle S_\rho(\bar{c}), R_1 \cap S_\rho(\bar{c})^{r_1}, \ldots, R_t \cap S_\rho(\bar{c})^{r_t}, c_1, \ldots, c_n \rangle$, where $\forall i$, R_i has arity r_i. Let $\approx$ denotes isomorphism of structures. We consider the equivalence relation $\approx_\rho$ on elements of a structure $\mathcal{G}$ where $\bar{a} \approx_\rho \bar{b}$ iff $N_\rho(\bar{a}) \approx N_\rho(\bar{b})$. Finally, let $ntp(d, \mathcal{G})$ be the number of equivalence classes of the relation $\approx_\rho$. We introduce the important notion of the locality rank of a query:

Definition 5. Given a query $\psi(u_1, \ldots, u_r)$, its locality rank is a number $\rho \in \mathbb{N}$ such that, for every $\mathcal{G} \in STRUCT[\tau]$ and two r-ary tuples $\bar{a}_1$ and $\bar{a}_2$ of $\mathcal{G}$, $N_\rho(\bar{a}_1) \approx N_\rho(\bar{a}_2)$ implies $\mathcal{G} \models \psi(\bar{a}_1) \Leftrightarrow \mathcal{G} \models \psi(\bar{a}_2)$. If no such ρ exists, the locality rank of ψ is ∞. A query is local if it has a finite locality rank. A language is local if each of its queries is local.

Gaifman's theorem [5] states for example that every first-order (relational calculus) query is local. The locality rank of a formula ψ is basically exponential in the depth of quantifier nesting in ψ, *but does not depend on the size of $\mathcal{G}$.*

As an example, we consider a graph instance $\mathcal{G} = \langle \mathcal{U}, R \rangle$ and the query $\psi(u, v) \equiv R(u, v)$ that enumerate all elements v at distance 1 of element u. This query has locality rank 1, i.e. it is sufficient to look at a neighborhood of radius 1 around u and v to devise if $\mathcal{G} \models \psi(u, v)$. Figure 3 shows $\mathcal{G}$ and neighborhoods $N_1(a)$ and $N_1(d)$ of elements a and d. Observe that there is 3 distinct (up to isomorphism) neighborhoods of radius 1, and that $N_1(a) \approx N_1(b), N_1(d) \approx N_1(e)$ and $N_1(c) \approx N_1(f)$.

We associate to each equivalence class of neighborhoods a unique number. Let $type(u)$ be a the number of the equivalence class of the neighborhood of u. In our example, $type(a) = type(b) = 1$, $type(d) = type(e) = 2$ and $type(c) = type(f) = 3$.

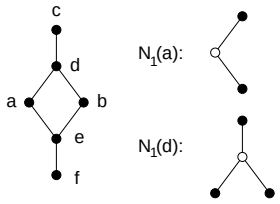

Figure 1: Instance and neighborhoods

Watermarking and locality. In the sequel we restrict our attention to structures in $STRUCT_k[\tau]$, i.e. structures with Gaifman graph of bounded degree k. Our aim is now to prove the following result:

THEOREM 3. *There exists a $(W^{1-q\varepsilon}, \frac{1}{\varepsilon}, d', \delta)$-marking - scheme preserving any local queries on $STRUCT_k[\tau]$, for the adversarial and non-adversarial model.*

To prove this theorem, we first observe that locality rank of queries implies a similarity between weighted elements involved in query computation, as shown is the following lemma.

LEMMA 1. *Let $\psi(u_1, \ldots, u_r, v_1, \ldots, v_s)$ be a formula with locality rank ρ, $\mathcal{G} \in STRUCT_k[\tau]$ and $\eta = (rk^{2\rho+1})^{\frac{1}{s}}$. Given $\bar{a}, \bar{b} \in \mathcal{U}^r$, $\bar{a} \approx_\rho \bar{b} \rightarrow |W_{\bar{a}} \backslash W_{\bar{b}}| \leq \eta$.*

PROOF. We prove it for $s = 1$. Let $\mathcal{G} \in STRUCT_k[\tau]$ and $\psi(\bar{u}, \bar{v})$ a query of locality rank ρ. Let $\bar{a}$ and $\bar{b} \in \mathcal{U}^r$, and $\bar{a} \approx_\rho \bar{b}$. Suppose that $|W_{\bar{a}} \backslash W_{\bar{b}}| > 2rk^{2\rho+1}$. So there exists an element $\bar{w} \in W_{\bar{a}} \backslash W_{\bar{b}}$ such that $\bar{w} \notin S_{2\rho+1}(\bar{a}, \bar{b})$, since $S_{2\rho+1}(\bar{a}, \bar{b})$ has at most $2rk^{2\rho+1}$ elements. By hypothesis, $\bar{w} \in W_{\bar{a}}$, so $\mathcal{G} \models \psi(\bar{a}, \bar{w})$. But since $\bar{w} \notin S_{2\rho+1}(\bar{a}, \bar{b})$, we know that $N_\rho(\bar{a}, \bar{w}) \approx N_\rho(\bar{b}, \bar{w})$. By locality of ψ, $\mathcal{G} \models \psi(\bar{b}, \bar{w})$, and $\bar{w} \in W_{\bar{b}}$. This contradicts the hypothesis. □

Figure 2 shows for any parameter u of ψ the set W_u and the isomorphism type of $N_1(u)$. Remark that although d and e have the same type, W_d and W_e are not identical, but differ on 2 elements.

u	$type(u)$	W_u					
		a	b	c	d	e	f
a	1				□	□	
b	1				□	□	
c	3				□		
d	2	□	□	□			
e	2	□	□				□
f	3					□	

Figure 2: types and active weighted elements

Let us consider watermarking for the previous example. It is worth noting that W_a and W_b are identical, i.e.

$$f_{(\mathcal{G},\mathcal{W})}(a, \psi) = f_{(\mathcal{G},\mathcal{W})}(b, \psi) = \mathcal{W}(d) + \mathcal{W}(e).$$

Suppose now that we modify the weight function $\mathcal{W}$ to a new function $\mathcal{W}'$, where $\mathcal{W}'(d) = \mathcal{W}(d) + 1$ and $\mathcal{W}'(e) = \mathcal{W}(e) - 1$. Then the distortion on function f is exactly zero, and query ψ is preserved by this 1-local distortion. The neutral transformation, where $\mathcal{W}'(d) = \mathcal{W}(d)$ and $\mathcal{W}'(e) = \mathcal{W}(e)$ yields of course the same property. Hence choosing one of these two distortions is an obvious way to hide one bit of information into this instance with global distortion 0 on a and b.

However, these distortions yields non-zero perturbations on other query results, e.g. for parameter c or f. The overall picture is given in figure 3.

u	$type(u)$	$mark$						$distortion$
					$+1$	-1		
		a	b	c	d	e	f	
a	1				□	□		0
b	1				□	□		0
c	3				□			+1
d	2	□	□	□				0
e	2	□	□				□	0
f	3					□		−1

Figure 3: mark and types

The general problem is then to find several distinct pairs of weighted elements to apply this (+1,-1) trick, with a restricted perturbation. With l such pairs, we can insert any boolean mark of l bits.

We briefly sketch the general technique used to find such pairs. We will first focus on *canonical parameters*, i.e. choose one representative parameter of each possible neighborhood. Because $\mathcal{G} \in STRUCT_k[\tau]$, there is a *finite* number $ntp(\rho, \mathcal{G})$ of such neighborhoods in the Gaifman graph of $\mathcal{G}$ (i.e. independent of the size of $\mathcal{G}$.) In our example, there are 3 different neighborhoods (or isomorphism types.)

We will then consider a partition of weighted elements into pairs, such that elements of each pair are in the query result set of *the same canonical parameters*. Using the preceding $(+1, -1)$ trick on these pairs guarantees a zero perturbation on queries with a canonical parameter.

To bound the distortion on any other possible parameter (not only canonical), we will apply lemma 1. A query with parameter of type i must depend on (almost) the same weights as the query of the corresponding canonical parameter. This limit the distortion to a constant. In order to get a watermarking scheme, where the distortion can be reduced at will, we will combine the previous technique with a randomized argument.

More formally, for all $i \in \{1, \ldots, ntp(\rho, \mathcal{G})\}$, let $\bar{a}_i \in \mathcal{U}^r$ be a tuple of type i, and $S = \{\bar{a}_1, \ldots, \bar{a}_{ntp(d,\mathcal{G})}\}$. We call S a set of *canonical parameters* for the query. Given a weighted element $\bar{w}$, its class $cl(\bar{w})$ is the set of isomorphism type of canonical parameters $\bar{a}_i$ such that $\bar{w} \in W_{\bar{a}_i}$. A partition $W^1, \ldots, W^n$ of a subset of W into pairs is said to be an S-partition if $\forall i$ and $W^i = (\bar{w}, \bar{w}'), cl(\bar{w}) = cl(\bar{w}')$.

Given a subset W' of $\{1, \ldots, n\}$, a *W'-pair marking* is the weight function $\mathcal{W}'$ such that $\forall \bar{w}, \mathcal{W}'(\bar{w}) = \mathcal{W}(\bar{w}) + m_{\bar{w}}$, where, for all $W^i = (\bar{w}_i, \bar{w}'_i)$:

- $i \in W' \rightarrow m_{\bar{w}_i} = +1, m_{\bar{w}'_i} = -1$.
- $i \notin W' \rightarrow m_{\bar{w}_i} = m_{\bar{w}'_i} = 0$.

Observe that the sum of distortion on each pair is always 0.

PROPOSITION 1. *Any W'-pair marking according to an S-partition is such that $\forall \bar{a}_i \in S, f_{(\mathcal{G},\mathcal{W})}(\bar{a}_i) - f_{(\mathcal{G},\mathcal{W}')}(\bar{a}_i) = 0$.*

PROOF. For $\bar{a}_i \in S$, $W_{\bar{a}_i} = W^{i_1} \oplus \cdots \oplus W^{i_j}$, where $\forall q, \forall \bar{w} \in W^{i_q}, i \in cl(\bar{w})$. Distortion on $W_{\bar{a}}$ is the sum of distortion on each pair, and each pair's distortion is 0. □

Figure 4 shows for our example canonical parameters, weighted elements and their classes, and a pair marking.

	$cl(\bar{w})$	2	2	2	1,3	1		
u	$type(u)$	$mark$						$distortion$
		$+1$	-1					
		a	b	c	d	e	f	
a	1				□	□		0
c	3				□			0
d	2	□	□	□				0

Figure 4: canonical parameters, classes and pair marking

A pair marking is said to be ε-good if it induces a global distortion smaller than $\frac{1}{\varepsilon}$. We will use a probabilistic distortion of weights, *according to a specific partition.*

PROPOSITION 2. *Let N be the number of distinct possible queries, and W' obtained by randomly choosing sets from an S-partition with probability $p = \frac{1}{\eta(2N)^\varepsilon}$. Then the W'-pair marking is an ε-good marking set of size $\Omega(p|W|)$ with probability at least $\frac{3}{4}$.*

PROOF. We first suppose that the S-partition can be chosen to cover a large part of W. For $\bar{a}$ as parameter, we consider the distortion induced on $f(\bar{a})$. If $\bar{a} \in S$, we apply proposition 1. Let $\bar{a} \notin S$ with type t, and $\bar{a}_t \in S$ its canonical parameter. Recall that

$$W_{\bar{a}_t} = W^{i_1} \oplus W^{i_2} \oplus \cdots \oplus W^{i_j}.$$

Since $\bar{a} \approx_\rho \bar{a}_t$, by lemma 1, $W_{\bar{a}}$ and $W_{\bar{a}_t}$ differ by at most η weights, i.e.

$$W_{\bar{a}} = ((W^{i_1} \oplus W^{i_2} \cdots \oplus W^{i_j}) \backslash A) \cup B,$$

where A are elements from $W_{\bar{a}_t}$ and B are elements not in $W_{\bar{a}_t}$, with $|A|+|B| \leq \eta$. The probability that $A \cup B$ contains at least $d = \lceil \frac{1}{\varepsilon} \rceil$ weights from W' is bounded by

$$\binom{|A \cup B|}{d} p^d \leq \eta^d p^d = \frac{1}{(2N)^{\varepsilon d}} \leq \frac{1}{2N}.$$

For such a marking W' and the corresponding structure $(\mathcal{G}, \mathcal{W}')$, we have for any $\bar{a}$:

$$\Pr_{\Omega}[|f_{(\mathcal{G},\mathcal{W})}(\bar{a}) - f_{(\mathcal{G},\mathcal{W}')}(\bar{a})| \geq \lceil \frac{1}{\varepsilon} \rceil] \leq \frac{1}{2N}.$$

Applying the union bound for all of the N possible queries to the previous equation, with probability at least $\frac{1}{2}$, global distortion is bounded by d on all queries. Furthermore, with probability at least $\frac{3}{4}$, $|W'| = \Omega(p|W|)$, using classical Chernoff bounds. Finally, if the S-partition does not cover W, we can apply the randomized technique of [10], proposition 4.3. □

We now prove our main theorem.

PROOF OF THEOREM 3. We begin with the non-adversarial model. By lemma 1, there is a constant q (independent of $|W|$) such that N is bounded by W^q. The marker generates random W' and checks until a ε-good marking W^* is obtained (each time, the distance is computed by considering all possible valuations for queries.) For any word m of length $l = p|W|$ as input, $(\mathcal{G}, \mathcal{W}_m)$ is returned, where $\mathcal{W}_m$ is the m^{th} ε-good marking corresponding to the m^{th} subset of W^*. The detector asks for weights described in W^* and outputs m. The marker performs $O(ntp(\rho, \mathcal{G})|\mathcal{U}^r|)$ isomorphism tests on constant size graphs, and generates $O(\ln(\eta N^\varepsilon))$ random bits. The detector checks $O(|W|)$ values by querying the suspect server. Notice that the marker needs only to find a random ε-good marking once, and can compute from it every watermarked instances.

Finally, this scheme follows Khanna and Zane's framework for the adversarial setting. Hence this watermarking scheme for the non-adversarial case can be turned into an adversarial scheme ([10], theorem 5.1.) □

REMARK 2. *For example, if $q = 30$ and if we consider that a distortion $\frac{1}{\varepsilon} = 40$ is acceptable, the amount of hidden bits is $|W|^{\frac{1}{4}}$. Hence, for a database with $|W| = 5000$ weighted elements, $5000^{\frac{1}{4}} = 8$ bits are hidden, hence $2^8 = 64$ different watermarked copies can be distributed. But q is related to the locality rank of queries, and can be rather huge for practical applications.*

4. PRESERVING MSO-QUERIES ON TREES AND TREE-LIKE STRUCTURES

In this section we consider the problem of watermarking labeled trees and tree-like structures, while preserving MSO-queries. These structures can easily model XML documents.

EXAMPLE 4. *This picture shows an XML document with a possible 1-local distortion. We also consider the following parametric Xpath query:*

$$\psi(a, v) = \texttt{school/student[firstname=a]/exam}$$

```
<school>
  <student>
   <firstname>John</firstname>
   <lastname>Doe</lastname>
   <exam>11</exam>
  </student>
  <student>
   <firstname>Robert</firstname>
   <lastname>Durant</lastname>
   <exam>16</exam>
  </student>
  <student>
   <firstname>Robert</firstname>
   <lastname>Smith</lastname>
   <exam>12</exam>
  </student>
</school>

<school>
  <student>
    <firstname>John</firstname>
    <lastname>Doe</lastname>
    <exam>11</exam>
  </student>
  <student>
    <firstname>Robert</firstname>
    <lastname>Durant</lastname>
    <exam>15</exam>
  </student>
  <student>
    <firstname>Robert</firstname>
    <lastname>Smith</lastname>
    <exam>13</exam>
  </student>
</school>
```

Then $f(\texttt{Robert}, \psi) = 28$ on the original document, and has distortion 1 on the second.

We will use Grohe and Turán notion of definability of a k-ary formula by a tree-automaton [7].

Trees and automaton-definable queries.

A binary tree is viewed as a $\{S_1, S_2 \preceq\}$-structure, where S_1, S_2 and $\preceq$ are binary relation symbols.
A tree $\mathcal{T} = \langle T, S_1^{\mathcal{T}}, S_2^{\mathcal{T}}, \preceq^{\mathcal{T}} \rangle$ has a set of nodes T, a left child relation $S_1^{\mathcal{T}}$ and right child relation $S_2^{\mathcal{T}}$. Relation $\preceq^{\mathcal{T}}$ stands for the transitive closure of $S_1^{\mathcal{T}} \cup S_2^{\mathcal{T}}$, i.e. the tree-order relation. A weighted tree $(\mathcal{T}, \mathcal{W})$ is a tree with a weight assignment $\mathcal{W} : T^s \to \mathbb{N}$. Given a finite alphabet Σ, let $\tau(\Sigma) = \{S_1, S_2, \preceq\} \cup \{P_c | c \in \Sigma\}$ where for all $c \in \Sigma$, P_c is a

unary symbol. A Σ-tree is a structure $\mathcal{T} = \langle T, S_1^{\mathcal{T}}, S_2^{\mathcal{T}}, \preceq^{\mathcal{T}}, (P_c^{\mathcal{T}})_{c\in\Sigma}\rangle$, where its restriction $\langle T, S_1^{\mathcal{T}}, S_2^{\mathcal{T}}, \preceq^{\mathcal{T}}\rangle$ is an ordered binary tree and for each $a \in T$ there exists exactly one $c \in \Sigma$ such that $a \in P_c^{\mathcal{T}}$. We denote this unique a by $\sigma^{\mathcal{T}}(a)$.

We consider trees with a finite number of distinguishable *pebbles* placed on vertices. For some $k \geq 1$, let $\Sigma_k = \Sigma \times \{0,1\}^k$. For a Σ-tree $\mathcal{T}$ and a tuple $\bar{a} = (a_1, \ldots, a_k)$ of vertices of $\mathcal{T}$, let $\mathcal{T}_{\bar{a}}$ be the Σ_k-tree with the same underlying tree as $\mathcal{T}$ and $\sigma^{\mathcal{T}_{\bar{a}}}(b) = (\sigma^{\mathcal{T}}(b), \alpha_1, \ldots, \alpha_k)$, where $\alpha_i = 1$ iff $b = a_i$.

A Σ-tree automaton is a tuple $\mathcal{B} = (Q, \delta, F)$. Set Q is a set of states, and $F \subseteq Q$ is a set of accepting states. Function $\delta : ((Q \cup \{*\})^2 \times \Sigma) \to Q$ is the transition function ($* \notin Q$.) A run $\rho : T \to Q$ of $\mathcal{B}$ on a Σ-tree $\mathcal{T}$ is defined as follows. If a is a leaf the $\rho(a) = \delta(*, *, \sigma^{\mathcal{T}}(a))$. If a has two children b_1 and b_2, then $\rho(a) = \delta(\rho(b_1), \rho(b_2), \sigma^{\mathcal{T}}(a))$. If a has only a left child b then $\rho(a) = \delta(\rho(b), *, \sigma^{\mathcal{T}}(a))$ and similarly if a has only a right child b, $\rho(a) = \delta(*, \rho(b), \sigma^{\mathcal{T}}(a))$. Finally, a Σ_{k+s}-tree automaton defines a s-ary query with k parameters $\mathcal{B}(\bar{a}, \mathcal{T}) = \{\bar{b} \in T^s : \mathcal{B} \text{ accepts } \mathcal{T}_{\bar{a}\bar{b}}\}$ on each Σ-tree $\mathcal{T}$. Let $W_{\bar{a}} = \mathcal{B}(\bar{a}, \mathcal{T})$.

It is well known that MSO-sentences and tree-automata have the same expressive power. For formula with free variables, a Σ_k-tree automaton is equivalent to an MSO-formula $\psi(u_1, \ldots, u_k)$ of vocabulary $\tau(\Sigma)$ if for all Σ-tree, $\mathcal{B}(\mathcal{T}) = \psi(\mathcal{T})$.

LEMMA 2 (GROHE,TURÁN [7]). *For any MSO-formula $\psi(u_1, \ldots, u_k)$ of vocabulary $\tau(\Sigma)$ there exists a Σ_k-tree automaton $\mathcal{B}$ that is equivalent to ψ.*

Preserving MSO-queries. Our final goal is now to prove the following theorem:

THEOREM 4. *There exists a watermarking scheme preserving any MSO-definable query on trees or classes of structures with bounded clique-width or bounded tree-width, in the adversarial and non-adversarial model.*

To prove this result, we first prove the following theorem, in order to apply lemma 2.

THEOREM 5. *There exists a $(\frac{|W|^{1-q\varepsilon}}{4m}, \frac{1}{\varepsilon}, d', \delta)$-marking-scheme preserving the query defined by a tree automaton with m states, in the adversarial and non-adversarial model.*

We begin by the following lemma:

LEMMA 3. *Let $\mathcal{B}$ be a Σ_2-tree automaton with m states. Then for every Σ-tree $\mathcal{T}$, there exists $n = |W|/4m$ distinct sets $V_1, \ldots, V_n \subseteq W$ and n distinct pairs $(b_i, b'_i) \in {V_i}^2$ of distinct weights such that $\forall i \neq j, V_i \cap V_j = \emptyset$, and $\forall a \in T$:*

$$a \notin V_i \to (b_i \in W_a \leftrightarrow b'_i \in W_a).$$

PROOF. We iterate a construct from [7]: from the bottom-up, we form $|W|/4m$ subtrees of $\mathcal{T}$ of size at least $2m$. Since the automaton has only m states, one can find in each V_i a pair of vertices such that the automaton ends in the same state on a given subtree, for all $a \notin V_i$.

More formally, from the bottom-up of $\mathcal{T}$, let U_1 be a minimal subtree with respect to inclusion with at least $2m$ elements. Since $\mathcal{T}$ is binary, U_1 contains at most $4m$ elements. We can repeat this construct $2n = \lfloor |T|/4m \rfloor$ times, obtaining sets $U_1, \ldots, U_{2n}$.

We consider the binary relation F on $H = \{U_1, \ldots, U_{2n}\}$ to be the set of all pairs (U_i, U_j) such that $lca(U_i) \prec^{\mathcal{T}} lca(U_j)$, and there is no k such that $lca(U_i) \prec^{\mathcal{T}} lca(U_k) \prec^{\mathcal{T}} lca(U_j)$. Then (H, F) is a forest with $2n$ vertices and at most $2n-1$ edges. Therefore there is at most n elements of this forest with more that 1 child. Without loss of generality, suppose that $U_1, \ldots, U_n$ have at most one child.

If U_i has no children, let $V_i = \{v \in T | lca(U_i) \preceq^{\mathcal{T}} v\}$, i.e. elements of the subtree of $\mathcal{T}$ rooted at $lca(U_i)$. If U_i has one child U_j, then let $V_i = \{v \in T | lca(U_i) \preceq^{\mathcal{T}} v \text{ and } lca(U_j) \not\preceq^{\mathcal{T}} v\}$, i.e. the set of all vertices of the subtree of $\mathcal{T}$ rooted at $lca(U_i)$ that are not in the subtree rooted at $lca(U_j)$. Observe that $V_1, \ldots, V_n$ are pairwise disjoint.

Let $1 \leq i \leq n$. If U_i has no child, then there exists two distinct elements $b_i, b'_i \in U_i$ such that:

- For all $a \notin V_i$, automaton $\mathcal{B}$ running on $\mathcal{T}_{ab_i}$ or $\mathcal{T}_{ab'_i}$ reaches $lca(U_i)$ in state q_i.

Now if U_i has a child U_j, and $q_1, \ldots, q_m$ are the states of $\mathcal{B}$, we define pairs $b_{i,k}, {b'}_{i,k}$ for $1 \leq k \leq m$ by induction on k. Suppose $1 \leq k \leq m$ and that $b_{i,l}$ and ${b'}_{i,l}$ are already defined for $l < k$. Since $|U_i| \geq 2m$ we have $|U_i \backslash \{b_{i,1}, \ldots, b_{i,k-1}\}| > m$. Therefore there exists distinct elements $b_{i,k}, {b'}_{i,k} \in U_i \backslash \{b_{i,1}, \ldots, b_{i,k-1}\}$ such that:

- There is a state $q_{i,k}$ of $\mathcal{B}$ such that if $a \notin V_i$, the automaton running on either $\mathcal{T}_{ab_i}$ or $\mathcal{T}_{ab'_i}$ and leaving $lca(U_j)$ is state q_k reaches $lca(U_i)$ in state $q_{i,k}$.

Finally, if U_i has no children, and $a \notin V_i$, $\mathcal{B}$ accepts $\mathcal{T}_{ab_i}$ if and only if $\mathcal{B}$ accepts $\mathcal{T}_{ab'_i}$. If U_i has one child U_j, $a \notin U_i$ and $\mathcal{B}$ ends in $lca(U_j)$ is state q_t, $\mathcal{B}$ accepts $\mathcal{T}_{ab_{i,t}}$ if and only if $\mathcal{B}$ accepts $\mathcal{T}_{ab'_{i,t}}$. □

We now claim that pairs (b_i, b'_i) are good candidates for a watermarking algorithm.

PROOF OF THEOREM 5. In the non-adversarial case, for a Σ_2-tree automaton, let $a \in T$. Suppose there is a j such that $a \in V_j$. Notice that in this case j is unique. Then for all $i \neq j$, $a \notin V_i$ and $\mathcal{B}$ accepts $\mathcal{T}_{ab_i}$ if and only if it accepts $\mathcal{T}_{ab'_i}$. Since distortion on weights b_i and ${b'}_i$ is zero, distortion on $f(a)$ is limited by the pair b_j, b'_j. This distortion is at most 1. Otherwise, if $\forall i, a \notin V_i$, then the induced distortion of all pairs is 0. This result generalizes to a Σ_{k+s}-tree automaton with the same randomized technique as proposition 2. For the adversarial case, we apply Khanna and Zane's transformation to the previous algorithm. □

We can now end with the proof of the main theorem.

PROOF OF THEOREM 4. For trees, applying lemma 2, we obtain an automaton $\mathcal{B}$ equivalent to ψ. Then, by theorem 5, there is a corresponding adversarial and non-adversarial watermarking scheme. To a structure $\mathcal{G}$ with bounded clique-width we can associate a labeled *parse-tree* $\mathcal{T}$. For any MSO-formula $\psi(\bar{u})$ there exists a MSO-formula $\tilde{\psi}(\bar{u})$ such that for $\mathcal{G}$ and the corresponding parse-tree $\mathcal{T}$, $\psi(\mathcal{G}) = \tilde{\psi}(\mathcal{T})$ (see [7], lemma 16.) Then by the previous remarks, there exists a watermarking scheme preserving $\tilde{\psi}$, hence ψ. Finally, structures with bounded tree-width k has clique-width at most 2^k, and the previous remark applies. □

Finally, we can state a converse to the previous result.

THEOREM 6. *There exists an MSO formula ψ and a class of structures with unbounded tree-width that do not possess a watermarking scheme preserving ψ.*

PROOF. Example 19 in [7] exhibits a MSO formula ψ with unbounded VC-dimension on the class of *grids*, which has unbounded tree-width. This shows actually that for all grid $\mathcal{G}$, the set $\bigcup_{\bar{a}} \psi(\bar{a}, \mathcal{G})$ is shattered by sets in $\{\psi(\bar{a}, \mathcal{G}) : \bar{a} \in \mathcal{U}^r\}$. The corresponding watermarking problem with the same formula ψ on the same class of structures is such that for all $\mathcal{G}, W^{\mathcal{G},\psi} = \bigcup_{\bar{a}} \psi(\bar{a}, \mathcal{G})$. Hence for all $\mathcal{G}$, its set of *active weighted elements* is shattered, so $VC(\psi, \mathcal{G}) = |W^{\mathcal{G},\psi}|$, showing by theorem 2 that no watermarking scheme is possible. □

5. INCREMENTAL WATERMARKING

In this section we suppose that a data owner needs to update the database and propagate changes to each of the registered data servers. The problem is then to maintain the watermark he has inserted.

Definition 6. For a class of updates U, a watermarking procedure/scheme $(\mathcal{M}, \mathcal{D})$ maintaining U is such that the same $\mathcal{D}$ is a detector for any database update in U.

Let $(\mathcal{G}, \mathcal{W})$ be a weighted instance. We consider first *weights-only updates*: the data server updates only the weighted part $\mathcal{W}$ while leaving $\mathcal{G}$ unchanged. In this case, updating the watermarked instance is easy.

THEOREM 7. *Previous watermarking schemes maintain weights-only updates.*

PROOF. For a weight distortion

$$\mathcal{W}'_0(\bar{a}) = \mathcal{W}_0(\bar{a}) + M,$$

in the original watermarked instance, and a new weight $\mathcal{W}_1(\bar{a})$, we propagate the *same distortion* M:

$$\mathcal{W}'_1(\bar{a}) = \mathcal{W}_1(\bar{a}) + M.$$

The same global distortion is obtained for the new instance. Since the detector extract the watermark by computing the difference between $\mathcal{W}(\bar{a})$ and $\mathcal{W}'(\bar{a})$, it is only sensitive to the modification M, and the watermark can be recovered. □

Now if updates modify $\mathcal{G}$, hence modify sets $W_{\bar{a}}$, bounded distortion is not guaranteed. This problem is harder than incremental updatability considered in [1]. The brute-force method that consists in computing a new watermarked version of the new database and distributing it is expensive, and moreover exposes the owner to *auto-collusion attacks* in the adversarial model. A server, receiving several successive versions of a database can remove the watermark by averaging numerical data. An important point is then to detect when an update operation requires the brute-force method. In the sequel, an update is said to be *type-preserving* if no isomorphism type has been created or suppressed by this update.

THEOREM 8. *In the non-adversarial model, there exists a $(|W|, \eta, 0, 0)$-marking* procedure $(\mathcal{M}, \mathcal{D})$ *for local queries on $STRUCT_k[\tau]$ maintaining any type-preserving update.*

PROOF. We restrict our ambition to a marking procedure (not a scheme) with constant distortion η. Observe that any pair-marking introduced by the algorithm of theorem 3 has distortion at most η on any $W_{\bar{a}}$ whose isomorphism type is in S. Since no new type is created, it is not needed to modify the mark, and the detector can still detect it. □

A note on relative error

Classical studies from the literature on approximation consider relative errors rather than absolute ones, because relative approximation is preserved under composition. Observe first that a relative perturbation $1 \pm \varepsilon$ of weights always yields a global distortion of at most $1 + \varepsilon$. Hence the watermarking problem becomes trivial. But relative error is not always appropriate because 1) for very small weights (close to 0), it induces a small and fragile perturbation, and 2) relative error does not necessarily model the problem we have in mind (mainly when error is less tolerable as weights increase.)

Conclusion and future work

In this paper we considered the problem of watermarking databases or XML documents, while preserving a set of queries in a specified language $\mathcal{L}$. We gave structural arguments for the existence of a watermarking scheme related to the VC-dimension of sets definable in $\mathcal{L}$. We showed that watermarking on arbitrary instances is impossible, and that languages and structures with bounded VC-dimension established by Grohe and Turán have also good watermarking properties. But we do not know if bounded VC-dimension is a sufficient condition to obtain a watermarking scheme.

Our model does not capture exactly the result from [10] since shortest path queries are indeed an *optimization problem* (notice however that the VC-dimension of weighted graphs with respect to their shortest path is bounded.) Optimization has received a large interest from the finite model theory community [11, 17]. An interesting point is to find relationships between logical definability of such problems, mainly their weighted versions [24], and their watermarking capacity.

Acknowledgments

I would like to express my gratitude to Michael Benedikt for his fruitful comments. I also thank Bernd Amann, Michel Scholl and Luc Segoufin for their precious discussions about this work, Julien Stern for initiating me into watermarking problems, Richard Lassaigne, Sylvain Peyronnet and several anonymous referees for their helpful remarks on this paper and its previous version.

6. REFERENCES

[1] R. Agrawal and J. Kiernan. Watermarking Relational Databases. In *International Conference on Very Large Databases (VLDB)*, 2002.

[2] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *J. of the Association for Computing Machinery*, 36(4):929–965, October 1989.

[3] I. J. Cox, M. L. Miller, and J. A. Bloom. *Digital Watermarking*. Morgan Kaufmann Publishers, Inc., San Francisco, 2001.

[4] J. Flum, M. Frick, and M. Grohe. Query evaluation via tree-decompositions. In *International Conference on Databases Theory (ICDT)*, volume 1973 of *Lecture Notes in Computer Science*, pages 22–38. Springer, 2001.

[5] H. Gaifman. On local and non-local properties. In *Proceedings of the Herbrand Symposium, Logic Colloquium'1981*, North Holland, 1982.

[6] M. Grohe and T. Schwentick. Locality of order-invariant first-order formulas. *ACM Transactions on Computational Logic (TOCL)*, 1(1):112–130, 2000.

[7] M. Grohe and G. Turán. Learnability and definability in trees and similar structures. *Lecture Notes in Computer Science*, 2285:645–658, 2002.

[8] S. Grumbach, L. Libkin, T. Milo, and L. Wong. Query language for bags: expressive power and complexity. *SIGACT News*, 27:30–37, 1996.

[9] S. Katzenbeisser and F. A. P. Petitcolas, editors. *Information hiding: techniques for steganography and digital watermarking*. Computer security series. Artech house, 2000.

[10] S. Khanna and F. Zane. Watermarking maps: hiding information in structured data. In *Symposium on Discrete Algorithms (SODA)*, 2000.

[11] P. Kolaitis and M. Thakur. Logical definability of NP optimization problems. *Information and Computation*, 115:321–353, 1994.

[12] L. Libkin and J. Nurmonen. Counting and locality over finite structures: a survey. In *Generalized Quantifiers and Computation*, Springer LNCS 1754, pages 18–50, 1999.

[13] L. Libkin and L. Wong. On the power of aggregation in relational query languages. In *Database Programming Languages (DBPL'97)*, Springer LNCS 1369, pages 260–280, 1997.

[14] L. Libkin and L. Wong. Query languages for bags and aggregate functions. *Journal of Computer and System Sciences*, 55(2):241–272, 1997.

[15] T. Milo, D. Suciu, and V. Vianu. Typechecking for XML Transformers. In *Symposium on Principles of Databases Systems (PODS)*, 2000.

[16] F. Neven and T. Schwentick. Query automata on finite trees. *Theoretical Computer Science*, 275:633–674, 2002.

[17] C. H. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. *Journal of Computer and System Sciences*, 43(3):425–440, 1991.

[18] G. Qu and M. Potkonjak. Hiding signatures in graph coloring solutions. In *Information Hiding*, pages 348–367, 1999.

[19] G. Qu, J. L. Wong, and M. Potkonjak. Optimization-intensive watermarking techniques for decision problems. In *DAC*, pages 33–36, 1999.

[20] R. Sion, M. Atallah, and S. Prabhakar. On watermarking semi-structures. Technical Report TR 2001-54, CERIAS, Nov 2001.

[21] L. Valiant. The complexity of enumeration and reliability problems. *SIAM Journal of Computing*, 8(3), 1979.

[22] L. G. Valiant. A theory of the learnable. In *Symposium on Theory of Computing*, pages 436–445, 1984.

[23] G. Wolfe, J. L. Wong, and M. Potkonjak. Watermarking graph partitioning solutions. In *DAC*, pages 486–489, 2001.

[24] M. Zimand. Weighted NP optimization problems: logical definability and approximation properties. *SIAM Journal of Computing*, 28(1):36–56, 1998.