4th Information
Hiding Workshop
Pittsburgh, PA, USA
25-27 April, 2001

Digital Watermarking of Chemical Structure Sets

Joachim J. Eggers[1], W.-D. Ihlenfeldt[2], and Bernd Girod[3]

[1] Telecommunications Laboratory, University of Erlangen-Nuremberg
Cauerstr. 7/NT, 91058 Erlangen, Germany, eggers@LNT.de
[2] Computer Chemistry Center, University of Erlangen-Nuremberg
Nägelsbachstr. 25, 91052 Erlangen, Germany, wdi@ccc.chemie.uni-erlangen.de
[3] Information Systems Laboratory, Stanford University
Stanford, CA 94305-9510, USA, girod@ee.stanford.edu

Abstract. The information about 3D atomic coordinates of chemical structures is valuable knowledge in many respect. For large sets of different structures, the computation or measurement of these coordinates is an expensive process. Therefore, the originator of such a data set is interested in enforcing his intellectual property right. In this paper, a method for copyright protection of chemical structure sets based on digital watermarking is proposed. A complete watermarking system including synchronization of the watermark detector and verification of the decoded watermark message is presented. The basic embedding scheme, denoted SCS (Scalar Costa Scheme) watermarking, is based on considering watermarking as a communications problem with side information at the encoder.

1 Introduction

Chemical structures are inherently three-dimensional, although most structure databases store them only as flat graphs. For many scientific studies, for example the development of drugs, the 3-D structure is a major factor determining the application potential of a compound. It is possible to determine 3-D atomic coordinates by experimental techniques, but this is very expensive. As an alternative, computational methods of various precision levels exist which take a structure graph or very rough 3-D structure approximation as input and compute 3-D atomic coordinates. For large datasets containing hundreds of thousands of molecules, quantum-chemical or fully optimizing force-field methods are not usable because they are too computationally expensive. Expensive optimizations can largely be avoided by model builders which employ complex rule-driven heuristics. The development of such programs is difficult, and represents a significant investment. Consequently, these programs are expensive when bought commercially, and coordinate sets, which are needed to isolate functional principles common among compounds with similar biological activity, represent a tangible value, even if the underlying structures are in the public domain. Due to the value of computed structure data, the originator is interested in enforcing the copyright of the data. Thus, robust labeling and identification of structure data is desired. Here, *digital watermarking* of the molecule structure data is investigated as one method for such labeling and identification. The intellectual property of the data set resides only in the atomic coordinates. Taking into account the limited precision of the model builder, a variation of the coordinates is acceptable and can be used for watermarking purposes. Given the small

size of typical records for one structure, it is certainly not possible to robustly mark every record, but this is not necessary. We are mainly interested in identifying the origin of large data sets, e.g., including 100,000-200,000 structures. Resistance against tampering by adding small amounts of random jitter to the coordinates, in addition to resistance against rotations and translations, is desirable. A more comprehensive list of possible attacks is given in Section 4.1.

Digital Watermarking has been investigated intensively during the last years in the context of multimedia data, e.g., audio, image or video data. Most blind watermarking techniques, where the watermark detector has no access to the original data, are based on spread-spectrum techniques, but recently much more powerful techniques have been proposed. One such method is called SCS (Scalar Costa Scheme) watermarking. SCS watermarking is appropriate for many different data characteristics, and thus is used here for embedding watermarks into the molecule data.

In Section 2, the basic principles and design criteria for SCS watermarking are reviewed. Next, the problem of detecting the existence of a SCS watermark is discussed in Section 3. In Section 4, the specific system design for SCS watermark embedding into and detection from the chemical structure data is described. The performance of the proposed scheme is investigated experimentally, and simulation results are presented in Section 5.

2 SCS Watermarking

We consider digital watermarking as a communication problem. The watermark encoder derives from the watermark message m (sometimes also called "payload") and the host data $\mathbf{x}$ an appropriate watermark sequence $\mathbf{w}$ which is added to the host data to produce the watermarked data $\mathbf{s}$. $\mathbf{w}$ must be chosen such that the distortion between $\mathbf{x}$ and $\mathbf{s}$ is negligible. Next, an attacker might modify the watermarked data $\mathbf{s}$ into data $\mathbf{r}$ to impair watermark communication. The attack is only constrained with respect to the distortion between $\mathbf{x}$ and $\mathbf{r}$. Finally, the decoder determines from the received data $\mathbf{r}$ an estimate $\hat{m}$ of the embedded watermark message. The encoder and decoder must be designed such that $\hat{m} = m$ with high probability. In *blind* watermarking schemes, the host data $\mathbf{x}$ are not available to the decoder. The codebook used by the watermark encoder and decoder is randomized dependent on a key K to achieve secrecy of watermark communication. Usually, a key sequence $\mathbf{k}$ is derived from K to enable secure watermark embedding for each host data element. Here, $\mathbf{x}$,$\mathbf{w}$,$\mathbf{s}$,$\mathbf{r}$ and $\mathbf{k}$ are vectors, and x_n,w_n,s_n,r_n and k_n refer to their respective nth elements.

Fig. 1 depicts a block diagram of blind watermark communication, where an attack by additive white Gaussian noise (AWGN) $\mathbf{v}$ is assumed. The depicted scenario can be considered communication with side information about the host signal at the encoder. For this scenario, Costa [3] showed theoretically that for a Gaussian host signal of power $\sigma^2_{\mathbf{x}}$, a watermark signal of power $\sigma^2_{\mathbf{w}}$, and AWGN of power $\sigma^2_{\mathbf{v}}$ the maximum rate of reliable communication (capacity) is $C = 0.5 \log(1 + \sigma^2_{\mathbf{w}}/\sigma^2_{\mathbf{v}})$, independent of $\sigma^2_{\mathbf{x}}$. The result is surprising since it shows that the host signal $\mathbf{x}$ need not be considered as interference at the decoder although the decoder does not know $\mathbf{x}$.

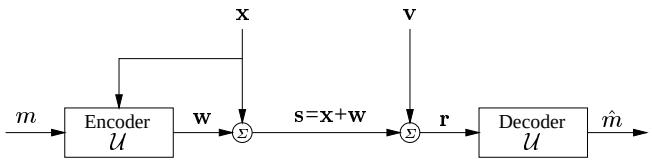

Fig. 1. Watermark encoding followed by an AWGN attack.

Costa's scheme involves a **random** codebook $\mathcal{U}$ which must be available at the encoder and the decoder. Unfortunately, for good performance the codebook must be so large that neither storing it nor searching it is practical. Thus, we proposed replacing it by a structured codebook, in particular a product codebook of dithered uniform scalar quantizers and called this scheme *SCS* (Scalar Costa Scheme) [4]. Note that SCS is very similar to Costa's original scheme, except for the suboptimal scalar quanitzer. The watermark message m is encoded into a sequence of watermark letters $\mathbf{d}$, where $d_n \in \mathcal{D} = \{0, 1\}$ in case of binary SCS. Note that this encoding process is usually divided into three steps. First, m is represented by a vector $\mathbf{u}$ with binary elements. Second, $\mathbf{u}$ is encoded into $\mathbf{u}_c$ by a binary error correcting code. Finally, $\mathbf{u}_c$ is mapped on $\mathbf{d}$ by selection or repetition of single coded bits so that each of the watermark letters d_n can be embedded into the corresponding host element x_n. The embedding rule for the nth element is given by

$$\begin{aligned} e_n &= \mathcal{Q}_\Delta \left\{ x_n - \Delta \left(\frac{d_n}{2} + k_n \right) \right\} + \Delta \left(\frac{d_n}{2} + k_n \right) - x_n \\ s_n &= x_n + \alpha e_n \end{aligned} \tag{1}$$

where $\mathcal{Q}_\Delta \{\cdot\}$ denotes scalar uniform quantization with step size Δ, and e_n is the error of subtractive dithered quantization. The key $\mathbf{k}$ is a pseudo-random sequence with $k_n \in [0, 1)$. The upper plot of Fig. 2 depicts one period of the PDF of the sent elements s conditioned on the sent watermark letter d_n and $k_n = 0$. The described embedding scheme depends on two parameters: the quantizer step size Δ and the scale factor α. Both parameters can be jointly optimized to achieve a good trade-off between embedding distortion and detection reliability for a given noise variance of an AWGN attack. Optimal values for Δ and α are given in [4]. In general, if accurate statistical models of the host data $\mathbf{x}$ are unavailable, and a MSE distortion measure is used, α and Δ can be designed for an AWGN attack with a specific watermark-to-noise power ratio (WNR). Note that this heuristic is only useful if a potential attacker does not have an accurate model for the host signal either.

At the decoder, the received data $\mathbf{r}$ is demodulated to obtain the data $\mathbf{y}$. The demodulation rule for the nth element is

$$y_n = \mathcal{Q}_\Delta \{r_n - k_n \Delta\} + k_n \Delta - r_n, \tag{2}$$

where $|y_n| \leq \Delta/2$. y_n should be close to zero if $d_n = 0$ was sent, and close to $\pm\Delta/2$ for $d_n = 1$. The lower plot in Fig. 2 shows the PDF of the demodulated elements y after

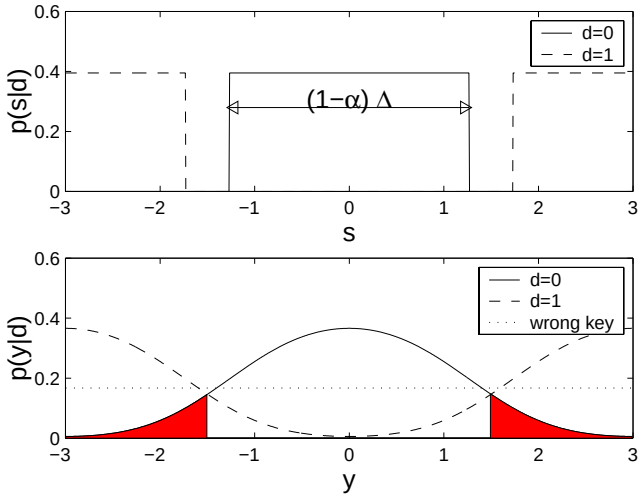

Fig. 2. One period of the PDFs of the sent and the received signal for binary SCS ($\sigma_{\mathbf{w}}^2$=1, WNR = 3dB, $\Delta = 6$, $\alpha = 0.58$). The filled areas represent the probability of detection errors assuming $d = 0$ was sent. The dotted line in the lower plot depicts the PDF when detecting with a wrong key $\mathbf{k}$.

AWGN attack conditioned on the sent watermark letter. $p_{\mathbf{y}}\left(y_n|d_n\right)$ can be computed numerically as described in [4]. In case of using an incorrect key $\mathbf{k}$ at the receiver, the distribution of $p_{\mathbf{y}}\left(y_n|d_n\right)$ will be uniform for any possible $\mathbf{r}$. This is indicated by the dotted line in the lower plot of Fig. 2.

The performance of SCS watermarking is discussed in detail in [4, 5]. It can be shown that for a large range of different WNRs SCS watermarking is superior to common blind spread-spectrum watermarking schemes since spread-spectrum watermarking suffers from large host signal interference. Note that the resiliency of SCS against AWGN attacks is independent from the host distribution. This property is particularly important for the application at hand, since the molecule coordinates of chemical structures do not have a smooth distribution, e.g., Gaussian or Laplacian, which is usually assumed in the design of detectors for spread-spectrum watermarks. It was also shown that at low watermarking rates, Spread Transform (ST) SCS watermarking is superior to SCS watermarking with simple repetition coding [5]. ST watermarking was originally proposed by Chen and Wornell [2] to improve binary dither modulation watermarking. In ST watermarking, the watermark is not directly embedded into the host signal $\mathbf{x}$, but into the projection $\mathbf{x}^{ST}$ of $\mathbf{x}$ onto a random sequence $\mathbf{t}$ of length τ. Any noise orthogonal to the spreading vector $\mathbf{t}$ does not impair watermark detection. Thus, an attacker, not knowing the exact spreading direction $\mathbf{t}$, has to introduce much larger distortions to impair an ST-SCS watermark than a simple SCS watermark. For AWGN attack and MSE distortion measurements, doubling the spreading length τ gives an additional power advantage of 3 dB for the ST-SCS watermark. Of course, this gain in detection reliability comes with a decrease of the watermark rate. In general, a spread transform of length τ

requires τ-times more data elements for watermark embedding. The optimal spreading length τ depends on the strength of attacks to be survived.

3 Verification of Decoded Watermark Information

So far, watermarking was considered as a communication problem where at the watermark decoder a watermark message $\hat{\mathbf{u}}$ is received assuming that a watermark is embedded with the key K. However, in many watermarking applications the detector has to decide whether a watermark with key K is embedded in the received data at all. Note that this problem differs somewhat from the communication problem.

For SCS watermarking we do not distinguish between the following cases:

- receiving non-watermarked data,
- receiving data that is watermarked with a different watermarking technique,
- receiving data being SCS-watermarked with a different key than the key K.

This is justified by the host signal independent nature of SCS watermark detection and the use of a key sequence $\mathbf{k}$ with being uniformly distributed in $[0, 1)$. Subsequently, we only distinguish between watermark detection from data watermarked with key K and from data not watermarked with key K.

We assume that SCS watermarking was designed to communicate the message $\mathbf{u}$ as reliably as possible via the watermarking channel. However, trying to detect an SCS watermark using a wrong key K, leads to demodulated data $\mathbf{y}$ that is uniformly distributed within $[-\Delta/2, \Delta/2)$ as indicated by the dotted line in Fig. 2. Thus, the decoded watermark message $\hat{\mathbf{u}}$ will be a random bit sequence with $p(\hat{u}_n = 0) = p(\hat{u}_n = 1) = 0.5$. The problem of deciding whether $\hat{\mathbf{u}}$ is a valid watermark message or just a random bit sequence can be formulated as a hypothesis test between

- hypothesis H_0: no watermark message is embedded in $\mathbf{r}$ with key K, and
- hypothesis H_1: the watermark message $\hat{\mathbf{u}}$ is embedded in $\mathbf{r}$.

In general, both hypotheses cannot be separated perfectly. Thus, we have to trade off the probability p_{FP} of accepting H_1 when H_0 is true (*false positive*) and the probability p_{FN} of accepting H_0 when H_1 is true (*false negative*).

Here, we devote a sub-vector $\mathbf{f}$ of length L_f of the watermark message $\mathbf{u}$ for verifying the validity of a received watermark message $\hat{\mathbf{u}}$. We compare two methods to decide between H_0 and H_1 using the verification bit vector $\mathbf{f}$. In our first approach, called method A, $\mathbf{f}$ is equal to the first L_f bits of $\mathbf{u}$ and error correction coding of $\mathbf{u}$ is such that the first L_{f_c} bits of the coded watermark message $\mathbf{u}_c$ are independent from the remaining watermark message bits. When detecting an SCS watermark letter from a data element where the embedded letter d_n is one of the coded verification bits $\mathbf{f}_c$, the probabilities for receiving a demodulated value y_n depending on hypothesis H_0 or H_1 are given as

$$p(y_n|\mathrm{H}_0) = \frac{1}{\Delta} \tag{3}$$

$$p(y_n|\mathrm{H}_1) = p_{\mathbf{y}}(y_n|d_n)\,. \tag{4}$$

Let $\mathcal{I}_f$ denote the index set of all data elements with embedded coded verification bits. Due to the independent identically distributed key sequence $\mathbf{k}$, the respective probabilities for detection from all data elements with index $n \in \mathcal{I}_f$ are given by

$$p\left(\mathbf{y}_{\mathcal{I}_f}|\mathrm{H}_0\right) = \prod_{n\in\mathcal{I}_f} p_{\mathbf{y}}\left(y_n|\mathrm{H}_0\right) \tag{5}$$

$$p\left(\mathbf{y}_{\mathcal{I}_f}|\mathrm{H}_1\right) = \prod_{n\in\mathcal{I}_f} p_{\mathbf{y}}\left(y_n|\mathrm{H}_1\right). \tag{6}$$

Applying Bayes' solution to the hypothesis test with equal a priori probabilities and equal costs for both hypotheses, H_1 is accepted if

$$R = \frac{p\left(\mathbf{y}_{\mathcal{I}_f}|\mathrm{H}_1\right)}{p\left(\mathbf{y}_{\mathcal{I}_f}|\mathrm{H}_1\right) + p\left(\mathbf{y}_{\mathcal{I}_f}|\mathrm{H}_0\right)} > 0.5. \tag{7}$$

Here, R, with $R \in [0,1]$, denotes the reliability with that a received watermark message $\hat{\mathbf{u}}$ is a valid watermark message.

In our second approach, denoted by method B, the verification message $\mathbf{f}$ is encoded together with all remaining watermark message bits to obtain the encoded watermark message $\mathbf{u}_c$. At the watermark receiver, the message $\hat{\mathbf{u}}$ is decoded as in the communication scenario. One part of $\hat{\mathbf{u}}$ is the decoded watermark verification message $\hat{\mathbf{f}}$ which must be identical to $\mathbf{f}$ for a valid watermark message $\hat{\mathbf{u}}$. Thus, the hypothesis decision rule is given by

$$\mathrm{H}_0 : \hat{\mathbf{f}} \neq \mathbf{f} \tag{8}$$

$$\mathrm{H}_1 : \hat{\mathbf{f}} = \mathbf{f}. \tag{9}$$

For both approaches, p_{FP} and p_{FN} are compared. For method A, p_{FN} and p_{FP} depend directly on the probabilities $p\left(\mathbf{y}|\mathrm{H}_0\right)$ and $p\left(\mathbf{y}|\mathrm{H}_1\right)$. Actual values for different detection cases will be given in Section 5. For method B, the false positive probability p_{FP} can be computed based on the assumption that $p\left(\hat{f}_n = 0|\mathrm{H}_0\right) = p\left(\hat{f}_n = 1|\mathrm{H}_0\right) = 0.5$. For L_f independent bits $\hat{f}_n$, we obtain $p_{\mathrm{FP}} = 0.5^{L_f}$. Thus, p_{FP} depends only on the number L_f of verification bits. The false negative probability depends on the bit error probability p_e and the number of verification bits and can be computed by $1-(1-p_e)^{L_f}$. Again, independent verification bits $\hat{f}_n$ are assumed. In practice, interleaving of all bits in $\mathbf{u}$ before error correction encoding is useful to ensure the validity of this assumptions as good as possible.

4 System Design for Watermarking of Chemical Structure Sets

In this section, the design of the entire watermarking system for chemical structure sets is described. First, possible attacks on watermarks in the structure sets are summarized. The watermarking system is designed such that the watermarks are as robust as possible against the mentioned attacks. An overview of all important design aspects is given and heuristic choices of system parameters are discussed.

4.1 Attacks on Chemical Structure Data

The type of attacks which can be envisioned for structure data sets is notably different from those applicable to audio-visual data and similar, classical areas of digital watermarking. First of all, raw watermarked structures can again be subjected to various different energy minimization procedures, including the algorithm which was initially used to generate the data, essentially re-computing the protected information. Protection against this type of attack is not possible. The initial information about the structural identity needs to be contained in the data file and can be used as basis for any further computation. However, we assume that no unlicensed copies of the software used to generate the original protected data are in circulation. Further, the computation time for larger datasets is often significant. Depending on the type of algorithm used and the size of the dataset, it can be up to several CPU months. Thus, simple re-generation of the data is often not a feasible approach. Attacks to remove or dilute the watermark are then limited to a small set of general, computationally inexpensive operations. These include:

- Removal of data from the original dataset, or injection of additional structures that are not watermarked, but possess coordinates from other, unmarked or differently marked sources.
- Re-ordering of the individual records in the dataset.
- Re-ordering of atoms and bonds in the structure records.
- Global 3-D transforms. Rotating or shifting the structures in 3-D space does not change their usability, since the intermolecular distances, angles and torsions define the characteristics of a molecule, not its orientation in 3-D space.
- Variation of structure notation. In some cases, structural features can be represented by different notational conventions without changing the identity of the structure. For instance, in a common format aromatic systems are represented as Kekulé systems. The sequence of single and double bonds can be re-arranged without changing the structure. These are comparatively simple operations, and all identification algorithms which use the structure as access key or generate canonic orderings of atoms will have to cope with this variability.
- Removal of atoms from structures. This operation is clearly a major modification of the structure, and the only case where the data retains at least a part of its usefulness is the global removal of hydrogen atoms.

4.2 Initial Considerations for the System Design

In general, a single structure does not contain enough data for an entire watermark message. Thus, the watermark message is distributed over several molecule structures, and watermark detection is only possible when several, perhaps modified, structures are available. The illegal use of single molecules cannot be proven, however, heavy illegal use of a large amount of the structure data should be detectable.

The watermark detector must have some information about the exact location of embedded watermark bits even after data re-ordering attacks as mentioned above. This problem is related to the well-known synchronization problem of watermark detectors.

Our system design is such that perfect synchronization of the watermark detector is always ensured. The required algorithms are described below.

The watermark message m is represented by a vector $\mathbf{u}$ of binary elements ($u_n \in \{0, 1\}$). $\mathbf{u}$ can include different information, e.g., an identifier of the copyright holder, a verification bit vector $\mathbf{f}$, and/or the date of computation of the molecule data. Further, the watermark embedding is dependent on a key K, which is only known to the copyright holder and perhaps a trusted third party.

The detection reliability may be improved by error correction codes. Thus, $\mathbf{u}$ is encoded into a binary vector $\mathbf{u}_c$ of length L_{u_c}. The influence of different error correction codes is investigated experimentally in Section 5. Note that only some of the L_{u_c} encoded watermark bits in $\mathbf{u}_c$ will be embedded into one molecule. Thus, decoding of $\hat{\mathbf{u}}$ must be possible even if some of the encoded bits $\mathbf{u}_c$ are not available from the data given at the watermark detector. To solve this problem, as much watermark information as possible is collected from each molecule, and this information has to be combined correctly to decode the watermark message $\hat{\mathbf{u}}$.

4.3 Structure Normalization and Hash Computation

In an attempt to embed or detect a watermark, the structure needs to be normalized and identified. Only parts of the encoded watermark message are embedded into each single structure (see Section 4.4). The specific message part to be embedded is determined by a hash code generated from the structure at hand. The hash code depends solely on the structure description. Thus, the watermark encoding and decoding process is independent of the order of structures in a large dataset. Also, insertion of unmarked records and deletion of marked records can be accepted to a comparatively high extent, since the additional or missing structures will only reduce the detection reliability, but no synchronization problems will ensue.

Hash codes for chemical structures being invariant to the operations mentioned in Section 4.1, exhibiting good randomness and negligible correlation in all bits, and do not generate hash code collisions for closely related structures, are not trivial, and have been studied extensively in chemoinformatics. We are using a state-of-the-art 64-bit hash code [7] which has some proven advantages over earlier attempts. Since the hash code depends on the hydrogen addition status, we always add a standard hydrogen set to the structure before computing the hash. If the hydrogen atoms are present, this is a null operation, otherwise new atoms are added with undefined or at least unmarked coordinates. These atoms, if added, will reduce detection reliability, but will ensure that the original structure hash code is regenerated and the original canonical atom order can be obtained.

Once the encoded message part has been identified with help of the hash code, the next preparation step is to move the structure to an unique 3-D orientation and generate a canonical ordering of the atoms. The canonical order of the atoms is determined by a symmetry-breaking sphere-expansion process. We use an adapted version of the Unique SMILES algorithm by Weininger et. al. [8]. This method is fast and exact for practically relevant structures. A few errors in re-establishing the precise atom order in highly symmetrical structures can be tolerated. We have enhanced the original algorithm to

include hydrogen atoms (whose coordinates are important) and to break some additional symmetric cases in a deterministic fashion.

4.4 Watermark Embedding into a Single Molecule Structure

The embedding of encoded watermark bits from $\mathbf{u}_c$ in the jth structure M_j is considered. A block diagram of the embedding scheme is depicted in Fig. 3. First, a canonic representation of the structure is obtained as described above. Next, the host data vector $\mathbf{x}_j$ is extracted (see also Section 4.7). Here, it is assumed that $L_{x,j}$ elements are extracted from the structure M_j, and the elements of $\mathbf{x}_j$ are scaled such that the watermark $\mathbf{w}_j$ can be embedded with a variance $\sigma^2_{\mathbf{w}} = 1$.

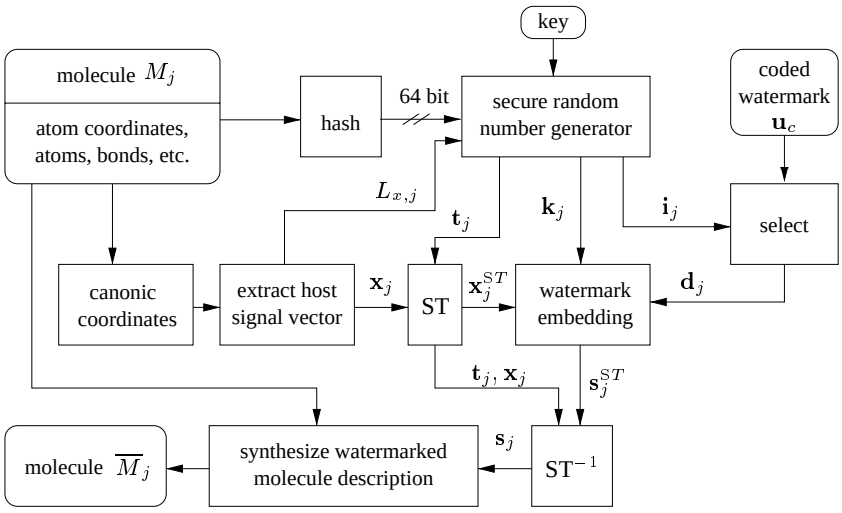

Fig. 3. Embedding of watermarks into the structure M_j.

For high detection reliability, it is useful to combine the watermark embedding scheme with a spread transform (ST) of length τ as discussed in Section 2. Thus, the host vector $\mathbf{x}_j$ of length $L_{x,j}$ is projected onto the random spreading vector $\mathbf{t}_j$ to obtain a vector $\mathbf{x}_j^{ST}$ of length $L_{x,j}^{ST} = \mathrm{floor}(L_{x,j}/\tau)$. Note that the ST reduces the number of bits that can be embedded into one structure.

A binary watermark letter $d_n \in \{0, 1\}$ is embedded into each element x_n^{ST} according to the embedding rule (1). Thus, $L_{x,j}^{ST}$ watermark letters can be embedded into each structure M_j. $L_{x,j}^{ST}$ differs a lot between different structures and is usually smaller than the length L_{u_c} of the encoded watermark message $\mathbf{u}_c$. Thus, $L_{x,j}^{ST}$ bits are pseudo-randomly selected from $\mathbf{u}_c$ to form the vector $\mathbf{d}_j$ of watermark letters to be embedded into $\mathbf{x}_j^{ST}$. The selected part of the encoded watermark message is determined by a pseudo-random index vector $\mathbf{i}_j$ where each index $i_{j,n} \in \{0, \ldots, L_{u_c} - 1\}$. Besides

$\mathbf{i}_j$, a pseudo-random key vector $\mathbf{k}_j$ with elements $k_{j,n} \in [0,1)$ is required to hide the embedded watermark to malicious attackers. The pseudo-random vectors $\mathbf{t}_j$, $\mathbf{i}_j$, and $\mathbf{k}_j$ must be perfectly reconstructible at the watermark detector and should not be known to unauthorized parties. Thus, the 64 bit hash value of the structure M_j is taken as seed for a cryptographic secure random number generator which is used to compute $\mathbf{t}_j$, $\mathbf{i}_j$ and $\mathbf{k}_j$ from this hash value dependent on the key K of the copyright holder. In the current implementation a pseudo-random number generator based on DES encryption is used.

The watermark letters $\mathbf{d}_j$ are embedded into $\mathbf{x}_j^{\mathrm{S}T}$ and the watermarked vector $\mathbf{s}_j^{\mathrm{S}T}$ is obtained. Finally, the inverse spread transform is applied to obtain $\mathbf{s}_j$ which is combined with the unmodified structure information to synthesize the watermarked molecule structure $\overline{M}_j$. Note that the embedding scheme is designed such that the watermark vector $\mathbf{w}_j = \mathbf{x}_j - \mathbf{s}_j$, describing the introduced modifications, has the variance $\sigma_{\mathbf{w}}^2 = 1$.

4.5 Watermark Detection from a Single Molecule

The upper part of the block diagram in Fig. 4 depicts the watermark detection scheme for one structure $\overline{\overline{M}}_j$. First, the data is transformed into its canonical representation. Next, the received vector $\mathbf{r}_j$ is extracted. The extraction method must be identical to the host vector extraction used for watermark embedding. Thus, the length of $\mathbf{r}_j$ is also $L_{x,j}$. Second, the 64-bit hash of $\overline{\overline{M}}_j$ is derived and the pseudo-random vectors $\mathbf{t}_j$, $\mathbf{k}_j$ and $\mathbf{i}_j$ are computed dependent on the copyright holders key K. After applying the spread transform, the demodulated soft watermark letters $\mathbf{y}_j$ are derived from $\mathbf{r}_j^{\mathrm{S}T}$ and $\mathbf{k}_j$ as described in Section 2. The probability $p\left(d_{n,j}=1\right)$ of receiving a watermark letter $d_{n,j}=1$ from the nth element of $\mathbf{y}_j^{\mathrm{S}T}$ is given by

$$p\left(d_{n,j}=1\right) = \frac{p_{\mathbf{y}}\left(y_{n,j}|d_{n,j}=1\right)}{p_{\mathbf{y}}\left(y_{n,j}|d_{n,j}=1\right) + p_{\mathbf{y}}\left(y_{n,j}|d_{n,j}=0\right)}. \tag{10}$$

These probabilities are collected in the vector $\mathbf{p}_{d_j}$. The required conditional probabilities $p_{\mathbf{y}}\left(y_{n,j}|d_{n,j}=0\right)$ and $p_{\mathbf{y}}\left(y_{n,j}|d_{n,j}=1\right)$ depend on the used watermarking scheme, but also on possible attacks. We designed our scheme for an AWGN attack of a certain noise variance, e.g. WNR $= -3$dB. This heuristic is useful since up to now little about possible statistical attacks on the watermarked structure data is known. The vectors $\mathbf{p}_{d_j}$ and $\mathbf{i}_j$ are the result of the detection process for the molecule $\overline{\overline{M}}_j$.

4.6 Joint Watermark Detection from Several Molecules

Assume that J structures $\overline{\overline{M}}_j$ are received, with $j \in \{0,\ldots,J-1\}$. The vectors $\mathbf{p}_{d_j}$ and $\mathbf{i}_j$ of length $L_{x,j}$ are derived as described above from each received structure. Further, we assume that the attack on the embedded watermark is memoryless, that is all demodulated watermark letters are statistically independent. Thus, the probability $p\left(u_{c,l}=1\right)$ that the lth coded watermark bit $u_{c,l}$ is 1, is given by

$$p\left(u_{c,l}=1\right) = \chi \prod_{\substack{j=0,\ldots,J-1 \\ n\,:\,l=i_{n,j}}} p\left(d_{n,j}=1\right). \tag{11}$$

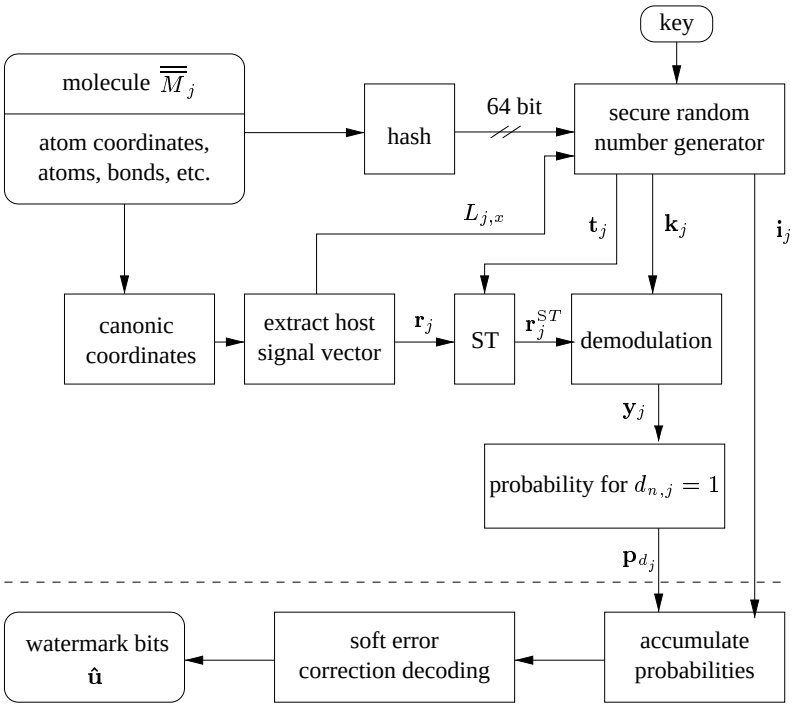

Fig. 4. Watermark detection from the structure $\overline{\overline{M}}_j$.

Here, $n : l = i_{n,j}$ denotes that the product is computed only for those probabilities $p\left(d_{n,j} = 1\right)$ where the corresponding index $i_{n,j}$ is equal to the index l of the considered coded watermark bit. χ is a constant such that $p\left(u_{c,l} = 1\right)$ is a valid probability.

Finally, a soft error correction decoding algorithm, e.g., a Viterbi decoder, is used to compute for $l = 0, \ldots, L_{u_c} - 1$, the most likely watermark message $\hat{\mathbf{u}}$ from the probabilities $p\left(u_{c,l} = 1\right)$. Note that it is possible for some l that no received data element $s_{n,j}$ is available. In this case, $p\left(u_{c,l} = 1\right)$ is initialized with 0.5, meaning $u_{c,l} = 1$ and $u_{c,l} = 0$ are equal probable.

4.7 Host Data Extraction and Quality Criteria

The host data vector $\mathbf{x}_j$ resembles the data of the structure M_j to be modified by the watermarking mechanism. Ideally, all elements of $\mathbf{x}_j$ are independent, such that watermarking one element does not affect the other ones. Further, it should be impossible for an attacker to derive the unwatermarked data $\mathbf{x}_j$ from the watermarked data $\mathbf{s}_j$. In the current version of our watermarking scheme, the host data contains the coordinates of all atoms. They are scaled such that a watermark of variance $\sigma_{\mathbf{w}}^2 = 1$ can be embedded without rendering the watermarked structure useless (see Section 5.4).

The quality of a 3-D structure dataset is measured by the energy (enthalpy of formation) of the conformers. Good coordinate generators will display a good balance between execution speed and conformer energy. The quality of a dataset can be checked by comparing the energy of the dataset structures to the energies obtained by using a more computationally expensive method to optimize the 3-D structures. Our primary test dataset was generated by the 3-D coordinate generator CORINA [6] which is very fast and employs only a low level of theory (rule-based initial coordinate generation and pseudo-force field energies for optimization). Since the testing of the acceptability of the watermarked structures requires a better level of theory than the original generator, we used the AM1 implementation of the VAMP package [1] which has been successfully used to process the same data set in a very expensive computational effort.

The acceptable level of distortion of the original coordinates depends on the precision of the original results. For CORINA coordinates, a change of 2-3% of the structure energy is tolerable. For an AM1 data set, less than 1% would be acceptable. For the CORINA dataset, we measured the compound energy before and after watermarking by performing a single-point AM1 computation which will not change and re-optimize the coordinates but only compute the energy of that coordinate set. In the current implementation, the modification of the atomic coordinates does not take into account the atomic environment at all. However, not all distortions of the structures lead to the same energy change. Thus, improved allocation of the watermark power to different coordinates should be investigated in the future.

5 Performance Evaluation

The described system for watermarking of chemical structure sets involves many different parameters, like the error correction code, the spread transform length τ, the watermark message length L_u, the parameter α and Δ for SCS watermarking, and the choice of verification bits $\mathbf{f}$. A detailed discussion of all parameters is beyond the scope of this paper. Here, we consider a watermark message $\mathbf{u}$ of fixed length $L_u = 96$ bits (equivalent to 12 ASCII characters). The parameter α and Δ were designed for an AWGN attack with WNR $= -3$dB. Thus, the SCS scheme was optimized for an AWGN attack where the power $\sigma^2_{\mathbf{v}}$ of additive noise $\mathbf{v}$ is twice as large as the watermark power $\sigma^2_{\mathbf{w}}$. Most of the experiments discussed below were performed on synthetic data since many simulations are required to measure low error probabilities. Nevertheless, some simulations results for chemical structure sets will be discussed, too.

5.1 Required Amount of Received Data Elements

The watermark bit error probability p_e was investigated experimentally for different amounts of received data elements. In practice, reliable detection from as few data elements as possible is desired. We restrict the discussion to an AWGN attack with WNR $= -3$dB. Rate 1/3 convolutional codes (CC) with memory $\nu = 4$ and $\nu = 9$ were used to encode all 96 watermark bits $\mathbf{u}$ into the coded bit vector $\mathbf{u}_c$ with length $L_{u_c} = 300$ and $L_{u_c} = 315$, respectively.

L_x random data elements $\mathbf{x}$ were chosen as host signal. This data was transformed into the spread transform domain where the projected data $\mathbf{x}^{\mathrm{ST}}$ has $L_x^{\mathrm{ST}} = L_x/\tau$ elements. Note that $\mathbf{x}^{ST} = \mathbf{x}$ for $\tau = 1$. For each element in $\mathbf{x}^{\mathrm{ST}}$, one bit of the encoded watermark message $\mathbf{u}_c$ was randomly selected and embedded. Simulations with 20000 random watermark messages were performed so that bit error probabilities about $p_e \approx 10^{-5}$ can be measured reliably. Fig. 5 shows the measured bit error probabilities p_e for CC with $\nu = 4$ and $\nu = 9$, and spread transform lengths $\tau = 1$ and $\tau = 4$. Obviously, the scheme with $\nu = 9$ and $\tau = 4$ performed best. Only 2000 data elements are required to achieve $p_e < 10^{-5}$. This corresponds to a watermark rate of about $1/10$ bit/element. About 1000 more data elements need to be received when using the less complex convolutional code with $\nu = 4$. Another 500 more data elements are required when leaving out the spread transform ($\tau = 1$).

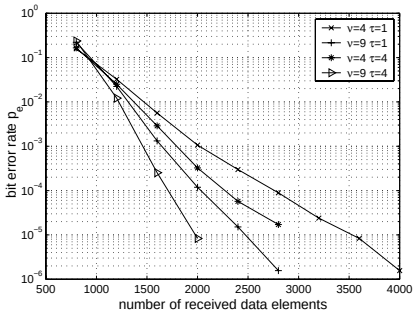

Fig. 5. Measured bit error probabilities for receiving 96 watermark message bits after AWGN attack with WNR = -3.0 dB. The watermark message was encoded with a rate 1/3 convolutional code with different memory length ν. Simulation results for spread transform lengths $\tau = 1$ and $\tau = 4$ are shown.

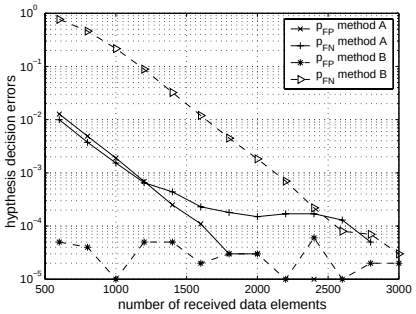

Fig. 6. False positive and false negative error probabilities for watermark verification. Two methods using 15 verification bits are compared. The watermarked data is attacked by AWGN with WNR$= -3$dB.

Note that the considered detection case is different from detection after a simple AWGN attack. The detection performance is impaired also by the randomness with which certain data elements are received. Simulation results show that lower error probabilities could be achieved when the number of embedding positions would be identical for all coded bits. However, in the application at hand, it is impossible to ensure that the watermark detector receives all watermarked data elements.

5.2 Verification of Decoded Watermark

Two methods for verifying the validity of a received watermark message were proposed in Section 3. Here, simulation results for both methods are compared. Fig. 6 shows the measured false positive and false negative probability for a verification bit vector $\mathbf{f}$ of length $L_f = 15$. The watermark message $\mathbf{u}$ was embedded with a rate $1/3$ CC with memory length $\nu = 4$.

The detection of 200000 random watermark messages was simulated and different amounts of received data was considered. The SCS parameter and channel noise were chosen as in the previous subsection. Hypothesis H_1 was valid in half of the cases, thus the error probabilities were estimated from 100000 decisions. For method B, a false positive probability $p_{\mathrm{FP}} = 0.5^{15} \approx 3 \times 10^{-5}$ can be expected. This value is verified by the simulation results shown in Fig. 6. The false negative error probability of method B depends on the bit error probability p_e which decreases for an increased number of received data elements. Fig. 6 shows that p_{FP} of method B also decreases slowly with the number of received data elements. Contrary, for method A the error probabilities p_{FP} and p_{FN} are almost identical when receiving few data elements. For an increased number of received data elements more false negative errors than false positive errors occur. Method B is superior with respect to the false positive rate when detecting from few data elements. However, the overall error probability is lower for method A. Note that for method A it also possible to achieve lower false positive rates by increasing the decision threshold which was 0.5 in (7). Of course higher false negative rates have to be accepted in such a case.

5.3 Perfect Attack on Parts of the Data

It is likely that an attacker has perfect knowledge about the original data for some part of the data set. In this case, the attacker simply replaces the watermarked data by the original data, thus erasing the watermark from the specific data elements. In general we found that reliable watermark detection can be achieved even for a substitution of 80% of the watermarked data elements. However, this is only possible when many data elements are available at the decoder. Thus, it is worth to select for the watermarking process only data elements which are unlikely to be known by an attacker. The disturbing influence of data replacement can be prevented this way.

5.4 Simulations with Example Molecule Data

Preliminary experiments with example molecule data were conducted. The host vectors $\mathbf{x}_j$ were composed by all atom coordinates of one molecule structure. The coordinate values were scaled by a factor of 1000 such that a watermark of power $\sigma_{\mathbf{w}}^2 = 1$ can be embedded. For this setting the AM1 energies in a 200-structure test set were changed by less than 0.3% on average, without producing outliers with unacceptable energies (more than 1.5% energy increase, corresponding to unusable structures). 25% of the structures were actually lower in AM1 energy after watermarking, demonstrating the imperfectness of the CORINA optimizer.

The watermark was detectable on this comparatively small dataset with near 100% confidence even after performing the following set of operations: Delete 10 random structures, add 10 similar structures without a watermark, re-compute unmarked coordinates for 10 random molecules, shuffle the sequence by moving 50 random structures into different slots and finally randomly rotate and translate all molecules. The algorithm proved to be very robust against this set of operations which we consider a typical smokescreen which could be applied by an attacker to conceal the origin of the data.

The detection of watermarks in hydrogen-depleted structures and the same set of operations is unreliable with the described host data extraction. The confidence value is only about 56% for the dataset with or without the additional smokescreen operations. These, as expected, do not have a measurable influence on the detection signal for this test case. This result indicates that it might be advantageous to embed the watermark only into the hydrogen-depleted structure representation.

6 Conclusion

A digital watermarking system for chemical structure sets was proposed. Watermarking was considered as a communication problem with side information at the encoder, where the watermark message is transmitted over an AWGN channel. Some bits of the watermark message are used for verifying the validity of a received watermark message. Two different methods for this validity check were proposed and compared. Both methods proved to be useful for watermark verification, however, differences in the false positive and false negative error probability have been found. One particularly interesting property of the proposed watermark detector is that watermark detection can be performed on any randomly selected sub-set of the watermarked data as long as this sub-set contains enough data elements. Any additionally received data element improves the detection reliability. Further, synchronization of the watermark detector can be ensured. For this, specific properties of the chemical structure sets are exploited.

References

1. B. Beck, A. Horn, J.E. Carpenter, and T. Clark. Enhanced 3-D databases: A fully electrostatic database of AM1-optimized structures. *Journal on Chemistry, Information and Computer Science*, 38:1214–1217, 1998.
2. B. Chen and G. W. Wornell. Achievable performance of digital watermarking systems. In *Proceedings of the IEEE Intl. Conference on Multimedia Computing and Systems*, volume 1, pp. 13–18, pages 13–18, Florence, Italy, June 1999.
3. M. H. M. Costa. Writing on Dirty Paper. *IEEE Transactions on Information Theory*, 29(3):439–441, May 1983.
4. J. J. Eggers, J. K. Su, and B. Girod. A blind watermarking scheme based on structured codebooks. In *Secure Images and Image Authentication, Proc. IEE Colloquium*, pages 4/1–4/6, London, UK, April 2000.
5. J. J. Eggers, J. K. Su, and B. Girod. Performance of a practical blind watermarking scheme. In *Proc. of SPIE Vol. 4314: Security and Watermarking of Multimedia Contents III*, San Jose, Ca, USA, January 2001.
6. J. Gasteiger, C. Rudolph, and J. Sadowski. CORINA. 3-D atomic coordinates for organic molecules. *Tetrahedron Comput. Method.*, 3:537–547, 1992.
7. W. D. Ihlenfeldt and J. Gasteiger. Hash codes for the identification and characterization of molecular structure elements. *Journal of Computational Chemistry*, 15:793–813, 1994.
8. D.A. Weininger and J. L. Weininger. SMILES 2. Algorithm for generation of unique SMILES notation. *Journal of Chemical Information and Computer Science*, 29:97–101, 1989.