Journal of Information Hiding and Multimedia Signal Processing ©2017 ISSN 2073-4212
Ubiquitous International Volume **8**, Number **6**, November **2017**

Audio Authenticity and Tampering Detection based on Information Hiding and Collatz p-bit Code

Diego Renza, Camilo Lemus and Dora M. Ballesteros L.

Telecommunications Engineering
Universidad Militar Nueva Granada
Carrera 11 101-80, Bogotá-Colombia
diego.renza@unimilitar.edu.co, lemus.camilo@gmail.com, dora.ballesteros@unimilitar.edu.co

Received May, 2017; revised July, 2017

Abstract. *A fragile watermarking algorithm for audio authenticity and tampering detection is presented. To generate the mark, a user-input text is used, which is spread by means of variable length binary codes obtained from Collatz conjecture. The embedding process is done in wavelet domain through quantization index modulation, and to mark the entire signal, a block repetition code is applied. The proposed method allows to identify the zones where the signal has been tampered with high accuracy. Experimental results show that the marked signal (i.e. protected audio signal) has high transparency, and allow tampering detection with high accuracy (overall accuracy above 99.5%, kappa index above 98%) and zero false alarms in most cases. Finally, as far as we know, the Collatz conjecture for authenticity purposes is used for the first time in this paper.*

Keywords: Fragile Watermaking, Collatz Conjecture, Audio authenticity, Tampering detection, Audio forensics.

1. **Introduction.** Modern technology, including hardware and software systems, provides accessible methods for producing, recording, editing, storing and distributing digital files. Several tools have been developed for both professional or non-professional purposes that make digital data can easily manipulated using desktop and mobile devices [1]. In some cases, like forensic analysis in legal field, it is necessary to prevent tampering of the evidence, i.e. to guarantee the Chain of Custody (CoC); hence it is needed the use of mechanisms to verify the originality of digital data and determine if it has been altered.

Methods for evaluating the authenticity of digital data can be classified into two groups: content-based identification methods and information hiding methods. The first group consists in extracting relevant features from the data and give a kind of digital signature of the digital data as a result [2]. The second group aims to imperceptibly insert useful information into digital data by means of watermarking, and, it is possible to extract that information later on to analyze the authenticity of the data [3].

In the first group, many of the traditional authenticity schemes are designed based on hash functions, due to the fact that they are suitable to summarize and verify large amounts of data. Hash functions ensure that content is unmodified, since they take a variable length message as an input and then compute every bit of the data stream to map it to an output message with fixed length, known as hash value or message digest [1]. An important property of hash functions is that they are extreme fragile, since a single bit flip is sufficient to change the digest and it will result in a completely different hash

value [2]. Nevertheless, some vulnerabilities have been discovered when algorithms such as some versions of message-digest algorithms (MD) and Secure Hash Algorithms (SHA) have been attacked [4, 5].

In information hiding methods, the data used can be a known mark of any class or associated to the digital data or its owner; in any case, differences between the embedded and retrieved mark indicate manipulation over the data [6]. The analysis and verification of the original and recovered marks aims to detect the alteration of the digital data, produced by malicious manipulations or unintentional modifications [7]. Here, the analysis of differences can be carried out using digital signal processing techniques and its goal is to establish if the evaluated digital data corresponds to the original data [8].

The methods of digital watermarking can be categorized as robust watermarking, semi-fragile watermarking and fragile watermarking [9]. While a robust watermark is designed to accomplish a high level of robustness against various attacks [10], a fragile watermark is designed to allow an easy destruction of the mark when slight changes are applied to the data [11]. Meanwhile, a semi-fragile watermark has a balance between robustness and fragility, i.e. it has good robustness against malicious manipulations but it is sensitive to classical user manipulations [12]. Regarding fragile watermarking for audio authenticity, some approaches have been proposed. In some cases, the mark is embedded in the transform domain, taking advantage of the frequency components of the signal to detect and locate manipulations [2, 1]. Another scheme consists in generating two marks from a hash function and speech sample points, and then, embed them into the wavelet coefficients [9]. The main limitation of such watermark schemes is related to guaranteeing that after the insertion process, the signal remains the same as the original signal (i.e. transparency of the output signal).

Although hash functions and watermarking techniques have similarities since they both can be used for the same purpose, there are also significant differences between them; for example, whereas watermarking techniques require modification of original content, hashing techniques require no previous modification; besides, using hashing techniques for audio authentication may require a database and, in addition, identification engines are needed to access and analyze this database; this is contrary to watermark detectors, which can operate independently [13]. Thirdly, in tampering detection, the hash functions give a binary answer, allowing to identify if the multimedia data have changed entirely or not, i.e. they cannot identify the places where the data have changed. On the other hand, fragile watermarking allows to obtain the locations where data have been modified [14]; in audio forensics cases, this feature is very useful since it can be important to detect the time ranges of the manipulations before the recording is used as evidence. In fourth place, whereas in cryptographic hash functions we have a hash value for each multimedia file, in watermarking it is possible to use the same signature (key) to mark different multimedia files.

According to the above, this paper presents a new fragile watermarking method for digital audio authenticity and tampering detection. To generate the mark, a user-input text is used, which is spread by means of variable length binary codes obtained from Collatz Conjecture; to mark the entire signal, redundancy is applied. The embedding process is done in wavelet domain through Quantization Index Modulation (QIM) with a low value of quantization step to increase the fragility.

Our contribution is three fold:

1. As far as we know, we propose the first method of fragile watermarking for audio authenticity based on the Collatz conjecture, in which a secret text is inserted into the voice signal.

2. Our method allows high accuracy of tampering detection, thanks to the proposed Collatz p-bit code which is unique for every of the 256 values.
3. Our method is high secured in terms of obscurity of the embedded bits, because the Collatz p-bit code has variable length, and every character of the secret text is mapped to a binary chain according to the Collatz p-bit code by using a random selection process, i.e. s/he cannot know the embedded string bits without having knowledge of the key.

2. Materials and methods.

2.1. **Discrete Wavelet Transform (DWT).** DWT provides a method for the multi-resolution analysis of signals at different scales with reliable time localization information [3]. The discrete wavelet transform has been widely used in many important signal processing applications including speech signal enhancement, watermarking, data hiding or steganography. The main objective of the DWT is to hierarchically decompose a signal into series of successively lower frequency approximation sub-bands and their respective detail ones [15]. To calculate the DWT of a signal x, two quadrature filters are used, h (low pass filter) and g (high pass filter), followed by a downsampling operation (Equations 1 and 2.

$$y_{low}[n] = \sum_{k=-\infty}^{\infty} x[k]h[2n-k] = (x * h) \downarrow 2 \tag{1}$$

$$y_{high}[n] = \sum_{k=-\infty}^{\infty} x[k]g[2n-k] = (x * g) \downarrow 2 \tag{2}$$

Here, the approximation coefficients y_{low} and detail coefficients y_{high} are the outputs of the h and g filters, respectively, after downsampling by 2 [16]. At each decomposition level, the time resolution is halved and the frequency resolution is doubled [3].

2.2. **Collatz Conjecture.** The Collatz Conjecture is a mathematical problem which asserts that starting from any positive number X, and applying the following sequence of operations, it always reaches 1: if X is even, divide it by 2; if X is odd, multiply it by 3 and add 1. Then repeat the process with the resulting number until the operation reaches 1 [17]. The conjecture was first proposed in early 1930s by Lothar Collatz. For more than 70 years, various researches have focused on verifying that from any arbitrary natural number, the sequence reaches 1 [18]. For small natural numbers, such as those used in the present work ($X \leq 256$), it is easy to check the validity of the conjecture. According to the above, the statement of the Collatz conjecture involves the iteration of the two following operations until $X = 1$ (Equation 3).

$$X = \begin{cases} X/2 & if \quad X \; is \; even \\ 3X+1 & if \quad X \; is \; odd \end{cases} \tag{3}$$

For example, if we apply Equation 3 and start with $X = 7$, the iteration goes: $7 \rightarrow 22 \rightarrow 11 \rightarrow 34 \rightarrow 17 \rightarrow 52 \rightarrow 26 \rightarrow 13 \rightarrow 40 \rightarrow 20 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$.

In section 2.3.1, the method to generate binary codes taking advantage of the Collatz Conjecture is shown.

2.3. **Proposed method for audio authenticity.** The proposed scheme for audio authenticity is based on the insertion of a binary code into the wavelet coefficients of the audio signal. It includes the modules of insertion and authentication.

2.3.1. *Insertion module.* The goal of this module is to insert a mark into an audio file for subsequent verification of authenticity. The process consists in inserting a secret alphanumeric text $(S(c), 1 \leq c \leq C)$ provided by the user, in the digital audio signal $(A(n), 1 \leq n \leq N)$; here, C is the number of characters of S, and N is the number of samples in A. S is a text that only can be known by the user who inserts the mark (e.g. legal authority), and A is the audio signal given as evidence. The values of L, delta and R (seed) are fixed in the system for both the insertion and authentication modules. The operation of this module is summarized in Figure 1.

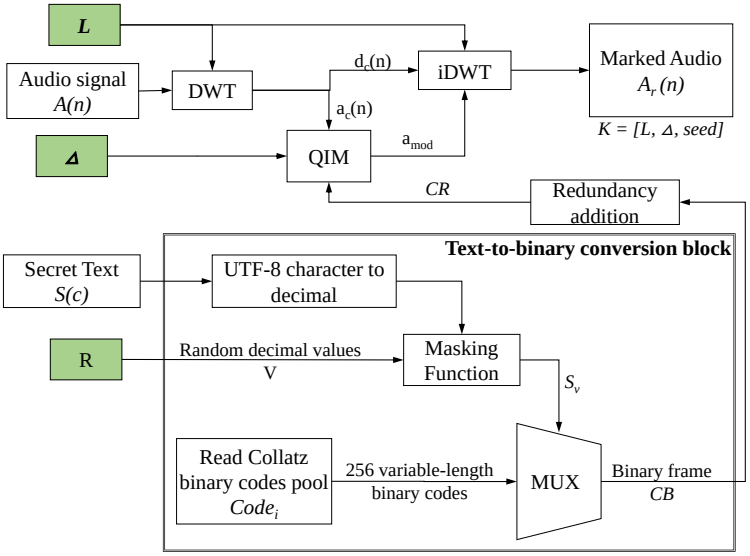


FIGURE 1. Block diagram of the proposed insertion module based on fragile watermarking.

First, a random vector (V) with the numbers in the range 1 to 256 is obtained. This vector is generated according to the value of seed (R). Then, each character of $S(c)$ is read as the decimal value of its corresponding UTF-8 code, and its value is masked using $S(c)$ as the index in the V vector, as shown in Equation 4.

$$S_v(c) = V(S(c)) \tag{4}$$

So far, the value of each element in the S_v vector will be the $S(c) - th$ position of V vector. Also, values of S_v are again in the range 1 to 256.

The next step consists in using a multiplexer, where each element in S_v is replaced by its correspondent Collatz binary code. Each value of S_v is used as the selection value of the multiplexer. The output is the $S_v - th$ row of the Collatz binary code pool, which is generated using the Collatz Conjecture.

Regarding the Collatz binary code pool generation, for each input value in the procedure, i.e. $i = [1, 2, \ldots 256]$, the Collatz binary code is generated by inserting a bit at the beginning of the code, depending on whether the value to be evaluated is even or odd. For even values, a bit 0 is inserted at the beginning of the code, otherwise, a bit 1 is inserted. In each iteration of the Collatz conjecture and using Equation 3, the generation of each Collatz binary code is given by Equation 5.

$$\begin{cases} if\ X\ is\ even \ \Rightarrow \ X = X/2 \ \ AND \ \ Code_i = [0 \ \ Code_i] \\ if\ X\ is\ odd \ \Rightarrow \ X = 3X + 1 \ \ AND \ \ Code_i = [1 \ \ Code_i] \end{cases} \tag{5}$$

The previous process is performed to represent each value in the S_v vector, as a p-bit Collatz binary code, where p is given by the number of iterations needed to reduce X to 1.

Applying Equation 5 to the example shown in section 2.2, where $X = 7$, the resultant Collatz binary code is '10000100010010101', and its length is given by the number of iterations needed to reduce the number to 1. The iterations and the Collatz binary code are as shown in Table 1. It is worth noting that the last iteration is always 1, whereby, the MSB (Most Significant Bit) of all codes is 1.

p	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
X	7	22	11	34	17	52	26	13	40	20	10	5	16	8	4	2	1
$Code_7$	1	0	1	0	1	0	0	1	0	0	0	1	0	0	0	0	1 (MSB)

TABLE 1. Example of the Collatz binary code obtaining.

Given the Collatz binary codes pool, $Code_i$, for $i = 1, 2, \ldots, 256$, and from the S_v values, a Collatz binary frame CB is obtained, as follows:

$$CB = [Code_{SV(1)}\ Code_{SV(2)}\ \ldots Code_{SV(C)}] \tag{6}$$

Here CB is a Collatz binary frame, obtained from the concatenation of the Collatz binary codes of each of the characters of the input text. The value of CB is the binary representation of the randomized input text.

In parallel, $A(n)$ is decomposed by means of DWT into approximation $a_c(n)$ and detail coefficients $d_c(l, n)$, where l is the level for the decomposition. To add a high degree of fragility, the selected component to embed the secret text is the approximation sub-band of the DWT; each coefficient of $a_c(n)$ will be used as the input sample in quantization function, and the details component, $d_c(l, n)$, will be preserved. Thus, the CB binary vector will be embedded into $a_c(n)$. To guarantee that the complete set of coefficients in $a_c(n)$ is marked, a block repetition code is used to create a redundant binary frame that is given by Equation 7.

$$CR = [CB_1 \quad CB_2 \quad \ldots \quad CB_i] \qquad for\ i\ =\ 1, 2, \ldots \lfloor \frac{N}{J \times 2^L} \rfloor \tag{7}$$

Where J is the number of bits in CB, calculated from Equation 6, L is the number of decomposition levels, and $\lfloor\ \rfloor$ is the integer part. As the first decomposition level is used, for each two samples of the audio host signal A, one bit of the binary code can be embedded. For the last samples of the signal, the code of each character is inserted one by one. In the worst case, the last 128 samples of the speech signal are not marked. However, this time slot is very small (e.g. 16 ms if frequency sampling is 8 kHz).

For embedding purposes, each bit in CR is used to quantize the respective $a_c(n)$ coefficient using the QIM method. Here, it is necessary to define a quantization step or delta value (Δ). The application of the QIM method involves using a quantization rule to hide a '0' and another rule to hide a '1'. In general, the quantized coefficients will belong to the data set $[0, \quad \Delta, \quad 2\Delta, \quad \ldots \quad n\Delta]$ when a '0' is hidden, and to the set $[\Delta/2, \quad 3\Delta/2, \quad \ldots \quad n\Delta/2]$ when a '1' is hidden. This is shown in Equation 8 [19].

$$a_{mod}(n) = \begin{cases} \Delta \lfloor \frac{a_c(n)}{\Delta} \rfloor & if \quad \text{CR(i)= 0} \\ \Delta \lfloor \frac{a_c(n)}{\Delta} \rfloor + \frac{\Delta}{2} & if \quad \text{CR(i)=1} \end{cases} \tag{8}$$

Lastly, the wavelet reconstruction process implies using the modified approximation sub-band, $a_{mod}(n)$ and the unmodified detail sub-band, $d_c(l, n)$, such as shown in Figure

1. Also L, Δ and the seed value for randomization make up the system key (K), which can be saved in a secure medium.

2.3.2. *Authentication module.* In order to assess the authenticity of the signal, a comparison between the secret and the recovered text has to be done. To recover the mark, the marked signal (Ar) and the secret key (K) are needed (Figure 2). The system has to check if the secret text matches all copies of the secret message, according to the redundancy block (Figure 2).

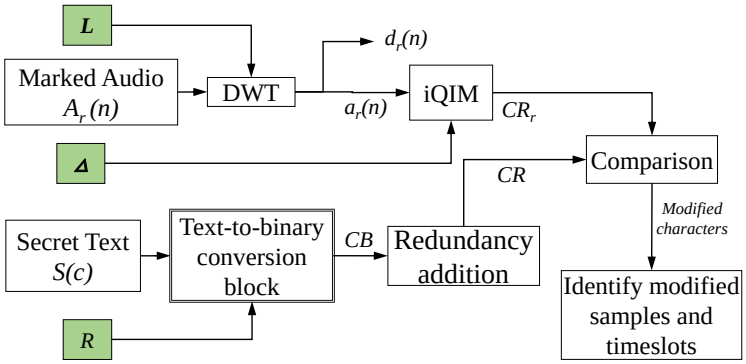


FIGURE 2. Block diagram of Recovering and verification of the mark.

The process shown in Figure 2 starts from the recovery of the binary data used in the Fragile marking of the audio file section. In this block, the secret text is converted to the binary representation with the same process of the "Text-to-binary conversion block" of the embedding module. The process starts from the same secret text ($S(c)$) and the same seed value to generate R vector. Its result is the CB vector (as defined in Equation 6). Also, a block repetition code is applied to obtain the CR vector (as defined in Equation 7). From the CB vector, the binary length data for both vector (J) and characters ($length_{code_c}$) are calculated and stored.

At the same time, the wavelet decomposition of the marked audio signal, A_r, is applied; here, the same filter, the same number of levels, and the same structure of the DWT of the insertion module are used (Figure 1). After decomposition, only the approximation sub-band ($a_r(n)$) is used, since it contains the information in its entirety. The binary information is then extracted using the extraction rules in QIM method. In these rules, the inputs are the same delta value of insertion process and each one of the coefficients in $a_r(n)$. The output of this phase is a binary CR_r vector, which is given by Equation 9.

$$CR_r(i) = \begin{cases} 1 & for \quad \frac{\Delta}{4} < |a_r(n) - \Delta\lfloor\frac{a_r(n)}{\Delta}\rfloor| \leq \frac{3\Delta}{4} \\ 0 & Otherwise \end{cases} \tag{9}$$

Then, a comparison process between CR_r and CR is done. The process consists in verifying that each pair of bits are equal. In case of finding any difference, the signal under test has been manipulated. Therefore, and taking into account that the recovered binary signal, CR_r, is a redundant vector, i.e. it has several copies of the Collatz binary code corresponding to the secret mark (CB), the next step consists in splitting the CR_r vector in blocks, whose size is the same as the CB vector. When using a Haar wavelet base, each copy of the CB_r block is obtained by means of the Equation 10.

$$CB_r(i) = \left\{CR_r(J \times i + j), \quad 1 \leq j \leq J, \quad 0 \leq i < \frac{N}{J \times 2^L}\right\} \tag{10}$$

Where J is the number of bits in CB, L is the number of decomposition levels, and N is the number of samples in A.

Again, a comparison to verify that each pair of bits are equal between each block of $CB_r(i)$ and CB is done. In case of finding any difference, the block position (i) is saved to later identify the places where data have changed. Besides, since the $CB_r(i)$ vector corresponds to the Collatz binary vector for the full secret mark, it is necessary to divide it in order to enhance the resolution in the tampering detection process. In this case, the comparison will be done character by character. Each recovered character, $Code_{rSV}(c)$ is given by Equation 11 and is compared with the corresponding $Code_{SV(c)}$ defined in CB vector (Equation 6).

$$Code_{rSV}(c) = \left\{ CB_{ri}(Pos_c + q), \quad 0 \leq q \leq length_{code_c}, \quad 1 \leq c < C \right\} \tag{11}$$

Where Pos_c is the relative position of the character within CB, calculated from the cumulative sum of the length of the codes $(length_{code_c})$.

Again, a comparison process between $Code_{rSV}(c)$ and $Code_{SV(c)}$ is done. This corresponds to the verification of each pair of characters, and in case of finding any difference between them, this will be the position where the signal under test has been manipulated. For tampering detection, the position of the character will be saved, so that in conjunction with the block position, it will be possible to identify the places where the signal has changed.

The above process is repeated until all characters in each block, and all the blocks have been compared.

In case blocks and marked characters have been detected, a process of tampering detection will be carried out. First, the initial (sb_{start}) and final (sb_{end}) samples of the marked block are calculated by means of Equations 12 and 13.

$$sb_{start} = (i-1) \times J \times L + 1 \tag{12}$$

$$sb_{end} = sb_{start} + J \times L - 1 \tag{13}$$

Where, i is the absolute position of the block within CR_r, J is the number of bits in CB, and L is the number of decomposition levels.

Then, the initial (sc_{start}) and final (sc_{end}) samples of the marked character are calculated by means of Equations 14 and 15.

$$sc_{start} = sb_{start} + pos_c \times L \tag{14}$$

$$sc_{end} = sc_{start} + length_{code_c} \tag{15}$$

Finally, using the sampling frequency of the signal, it is possible to obtain the timeslots where the modifications have been done.

In case there are no blocks or characters marked, it means the signal under test has not been manipulated.

3. **Results and discussion.** In order to evaluate the transparency, the fragility and the tampering detection accuracy of the proposed method, the following database was used:

- Number of audio files: 20 speech files. The files are monophonic, with lengths of 15, 30, 60, 90 and 120 seconds (4 files for each length), sampling frequency of 8 kHz or 48 kHz.
- Number of marks (secret texts): 4, with length of 2, 4, 8 and 16 characters.

Below, the data and tests used in each case are explained along with the obtained results.

3.1. **Transparency.** Transparency refers to the level of similarity between the marked signal and the original signal. This property is of great importance in audio forensics cases for authenticity verification, where a recording given as evidence is susceptible to be marked and therefore, its content after watermarking must be perceptually equal to the original recording and without significant distortions [20].

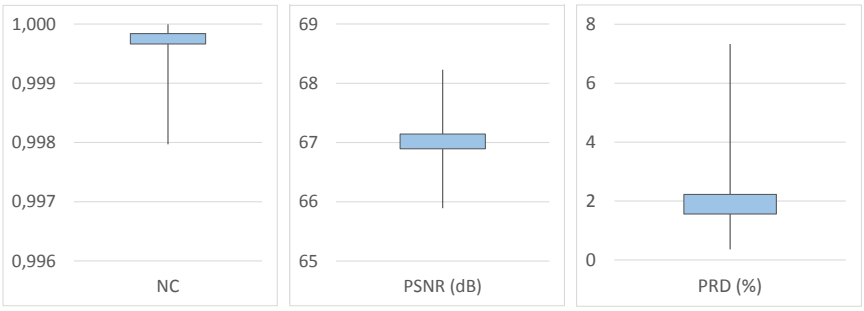
In order to estimate quantitatively the similarity between the original signal and the marked one, each mark is inserted into each audio file, giving as a result a total of 80 tests (i.e. 20 speech signals × 4 texts). Then, the three following metrics are applied: correlation coefficient (NC), peak signal to noise ratio ($PSNR$) and percent root-mean-square difference (PRD). The latter is defined in Equations 16.

$$PRD = \sqrt{\frac{\sum_i [A_i - B_i]^2}{\sum_i A_i^2}} x100 \tag{16}$$

Where A is the original signal, B is the watermarked signal, $\bar{X} = mean(X)$, $peakval$ is the maximum value that can take the signal, and MSE is the mean squared error.

The NC index measures the similarity of tendency in the local context of two signals; it ranges from -1 to 1, where 1 is the ideal value. The $PSNR$ index evaluates the fidelity of the signal representation regarding the original signal; it ranges from 0 dB to infinite, the higher its value, the lower the distortions. As PRD values are percentages, they range from 0 to 100, being 0 its ideal value.

Figure 3 shows the results of the 80 tests; these graphics show the summary results using confidence ranges, where the boxes represent the 95% of the data. The Figures show that the distortion of the signal is minimal, since the obtained values for the three metrics mentioned above are close to the ideal value. For the correlation coefficient, the NC obtained values are higher than 0.9996, for $PSNR$ the values are higher than 65 dB, whereas PRD are lower than 7.5%. From the obtained values, it is guaranteed that the quality of the audio recording is not degraded after the watermarking process, and therefore the watermarked signal could be used as evidence in legal cases.



(a) Correlation coefficient (b) PSNR (c) PRD

FIGURE 3. Confidence ranges (95%) for the imperceptibility results between the watermarked audio files and their original audio files. Results for the 80 tests.

3.2. **Fragility.** As discussed in section 1, a fragile mark is inserted in such a way that slight changes of the watermarked signal destroy the mark. Mute modification is selected as the signal processing operation to test the fragility. For this purpose, the 80 resulting marked audio files from the transparency tests are used. Each file is attacked with 2, 4 or 8 modifications of 1-second every one, giving as a result 240 tests. After attacking the signal, the authentication of the mark is performed by applying the procedure described in Section 2.3.2. In all cases, the modification of the signal was detected. Regarding the samples detected as modified, in each case their accuracy was evaluated with respect to the samples actually modified, which will be explained below.

3.3. **Tamper detection accuracy.** In each one of the 240 tests mentioned above, it is possible to calculate the number of samples detected as modified, and compare them with the samples corresponding to the timeslots actually modified. Therefore, it is possible to detect unchanged samples erroneously categorized as changed (false positives, FP), changed samples erroneously categorized as unchanged (false negatives, FN), changed and unchanged samples correctly identified (true positives, TP and true negatives, TN respectively). Using the latter, CC is expressed as $CC = TP + TN$ and using the first, $WC = FP + FN$, where CC means correct classification and WC wrong classification. Then, Overall Accuracy (OA) and Kappa (κ) indices are calculated to assess the accuracy in the tampering detection process (Equations 17, 18). OA ranges from 0 to 1, whereas kappa index ranges from -1 to 1.

$$OA = \frac{CC}{T} \quad with \quad T = CC \; + \; WC \tag{17}$$

$$\kappa = \frac{OA \; - P_e}{1 \; - P_e} \tag{18}$$

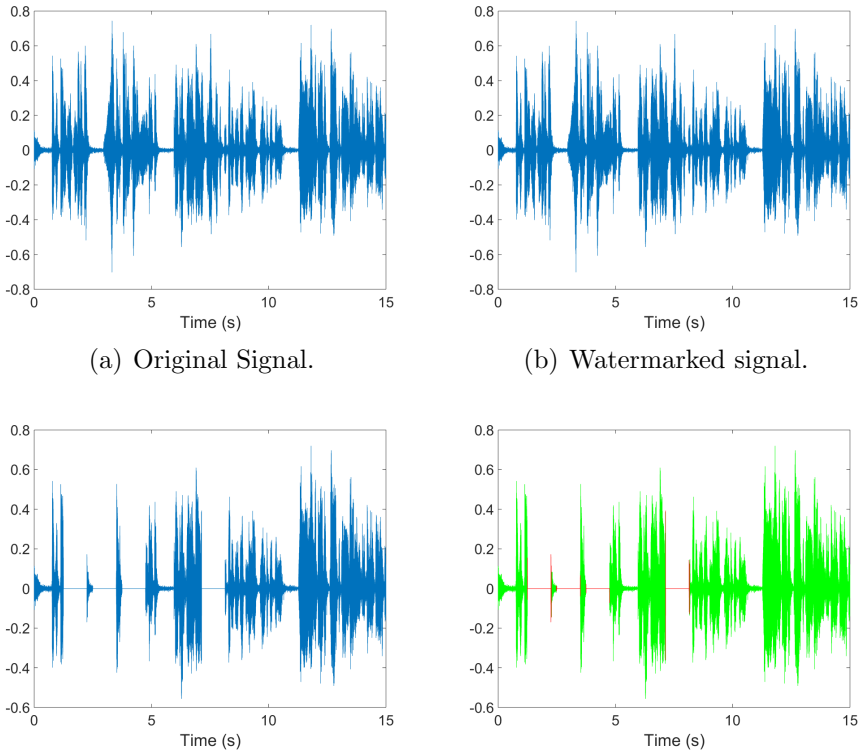
With, $P_e = \{P_1 \; * P_2\} \; + \{(1 \; - \; P_1) \; * \; (1 \; - \; P_2)\}$.

Where P_1 is the number of samples categorized as changed divided by the total number of samples of the watermarked file (i.e. $P1 = (TP + FP)/T$). P_2 is the real number of changed samples divided by the total number of samples of the watermarked file (i.e. $P_2 = (TP + FN)/T$).

An example of the performance of this algorithm for tamper detection is featured below. In this case, the signal used has a sampling frequency (f_s) of 8 kHz and 15 seconds of duration, for a total of 120000 samples. The text 'UMNGGNMU' is used to mark the audio signal. Four 1-second timeslots were modified, which is equivalent to 32000 samples. The modified timeslots correspond to (time in seconds): [1.25:2.25], [2.5:3.5], [3.75:4.75] and [7.14:8.14]. Figure 4 shows the resulting signals. Figure 4(a) shows the original signal, Figure 4(b) shows the watermarked signal, the attacked signal is shown in Figure 4(c), whereas the result of the tamper detection module is shown in Figure 4(d); here, the plot of tamper detection result shows samples detected as unmodified in green and samples detected as modified in red.

In the tampering detection process, the obtained modified timeslots are [1.25:2.25], [2.50:3.50], [3.75:4.75] and [7.14:8.14] seconds. In the detection process, the accuracy values are TP=32000, TN=87508, FP=492 and FN=0. The corresponding values of OA and κ index are 0.9959 and 0.989568, respectively. Regarding transparency, the results are PSNR=66.77 dB, NC=0.999993 and PRD=0.43%.

Finally, Figures 5(a) and 5(b) show the accuracy results for the 240 tests by means of radar plots. In these graphics, the radius represents the data and the angle the test number. As shown, most of the OA results are concentrated around 0.999 and its minimum

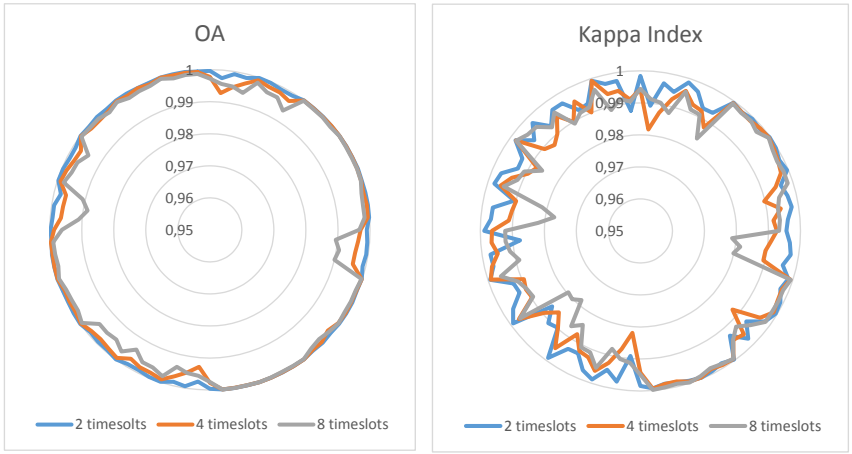


(a) Original Signal. (b) Watermarked signal.

(c) Attacked signal. (d) Tamper Detection.

FIGURE 4. Example of detection.

value is about 0.988. Regarding the κ index, most of the values range about 0.99 and the minimum one is approximately 0.977.



(a) Overall accuracy. (b) Kappa Index.

FIGURE 5. Detection accuracy. Radar plots for overall accuracy.

4. **Conclusion.** We proposed a scheme for audio authenticity based on the Collatz Conjecture through a bit-code generator that spreads a text input value. Our method shows very high results in terms of the transparency of the marked signal (i.e. evidence) and the accuracy in tamper detection. Regarding transparency, three parameters were used between the original and marked signals; in all cases, it was confirmed that the marked audio

signal does not have perceptual distortions. In addition, the performance in the detection of the tampered zones was evaluated, the modified timeslots were correctly identified in all cases, and the value of false negatives was zero in most cases. This results in a correct identification of manipulations, as well as the correct identification of no-manipulations, a useful condition for authenticity in the field of audio forensics.

Acknowledgment. This work is supported by the "Universidad Militar Nueva Granada-Vicerrectoría de Investigaciones" under the grant IMP-ING-2136 of 2016.

REFERENCES

[1] S. Zmudzinski and M. Steinebach, Perception-based audio authentication watermarking in the time-frequency domain, in *International Workshop on Information Hiding*, pp. 146–160, Springer, 2009.

[2] A. T. Ho and S. Li, *Handbook of digital forensics of multimedia data and devices*. Wiley-IEEE Press, 2015.

[3] Y. Lin and W. H. Abdulla, *Audio Watermark: A Comprehensive Foundation Using MATLAB*. Springer, 2014.

[4] S. R. Blackburn, D. R. Stinson, and J. Upadhyay, On the complexity of the herding attack and some related attacks on hash functions, *Designs, Codes and Cryptography*, vol. 64, no. 1-2, pp. 171–193, 2012.

[5] T. Xie, F. Liu, and D. Feng, Fast collision attack on md5., *IACR Cryptology ePrint Archive*, vol. 2013, p. 170, 2013.

[6] D. Renza, D. M. Ballesteros, and H. D. Ortiz, Text hiding in images based on qim and ovsf, *IEEE Latin America Transactions*, vol. 14, no. 3, pp. 1206–1212, 2016.

[7] P. Cano, E. Batlle, E. Gómez, L. de CT Gomes, and M. Bonnet, Audio fingerprinting: concepts and applications, in *Computational intelligence for modelling and prediction*, pp. 233–245, Springer, 2005.

[8] D. P. N. Rodríguez, J. A. Apolinário, and L. W. P. Biscainho, Audio authenticity: Detecting enf discontinuity with high precision phase analysis, *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 3, pp. 534–543, 2010.

[9] Q. Qian, H.-X. Wang, Y. Hu, L.-N. Zhou, and J.-F. Li, A dual fragile watermarking scheme for speech authentication, *Multimedia Tools and Applications*, pp. 1–20, 2015.

[10] A. G. Acevedo, Audio watermarking: properties, techniques and evaluation, *Information Security and Ethics: Concepts, Methodologies, Tools, and Applications*, vol. 6, pp. 23–61, 2007.

[11] J.-L. Liu, D.-C. Lou, M.-C. Chang, and H.-K. Tso, A robust watermarking scheme using self-reference image, *Computer Standards & Interfaces*, vol. 28, no. 3, pp. 356–367, 2006.

[12] W.-C. Chen and M.-S. Wang, A fuzzy c-means clustering-based fragile watermarking scheme for image authentication, *Expert Systems with Applications*, vol. 36, no. 2, pp. 1300–1307, 2009.

[13] J. Haitsma, T. Kalker, and J. Oostveen, Robust audio hashing for content identification, in *International Workshop on Content-Based Multimedia Indexing*, vol. 4, pp. 117–124, Citeseer, 2001.

[14] J. Serra-Ruiz and D. Megías, A novel semi-fragile forensic watermarking scheme for remote sensing images, *International journal of remote sensing*, vol. 32, no. 19, pp. 5583–5606, 2011.

[15] M. Zhao, J.-S. Pan, and S.-T. Chen, Optimal snr of audio watermarking by wavelet and compact pso methods, *Journal of Information Hiding and Multimedia Signal Processing*, vol. 6, no. 5, 2015.

[16] X.-F. L. Ming-Qian Wang and W.-J. Gu, A dwt-based covert timing channel of high concealment, *Journal of Information Hiding and Multimedia Signal Processing*, 2016.

[17] Ş. Andrei and C. Masalagiu, About the collatz conjecture, *Acta Informatica*, vol. 35, no. 2, pp. 167–179, 1998.

[18] T. O. e Silva, Empirical verification of the 3x+ 1 and related conjectures, *The Ultimate Challenge: The*, pp. 189–207, 2010.

[19] B. Chen and G. W. Wornell, Quantization index modulation: a class of provably good methods for digital watermarking and information embedding, *IEEE Transactions on Information Theory*, vol. 47, no. 4, pp. 1423–1443, 2001.

[20] T.-S. Wu, H.-Y. Lin, W.-C. Hu, and Y.-S. Chen, Audio watermarking scheme with dynamic adjustment in mute period, *Expert Systems with Applications*, vol. 38, no. 6, pp. 6787–6792, 2011.