Short 3-Secure Fingerprinting Codes for Copyright Protection*

Francesc Sebé and Josep Domingo-Ferrer

Universitat Rovira i Virgili, Dept. of Computer Engineering and Mathematics,
Av. Països Catalans 26, E-43007 Tarragona, Catalonia, Spain
{fsebe,jdomingo}@etse.urv.es

Abstract. A construction is presented to obtain 3-secure fingerprinting codes for copyright protection. Resistance against collusions of up to three buyers is achieved with a codeword length dramatically shorter than the one required by the general Boneh-Shaw construction. Thus the proposed fingerprints require much less embedding capacity. Due to their very clandestine nature, collusions tend to involve a small number of buyers, so that there is plenty of use for codes providing cost-effective protection against collusions of size up to 3.

Keywords: Electronic copyright protection, Fingerprinting, Watermarking, Buyer collusion.

1 Introduction

Successive failure of copy prevention systems has caused copy detection systems to become the most promising option to protect the intellectual property of multimedia content. In copy detection, the merchant embeds an imperceptible mark into the content before selling it. There are two kinds of mark: watermarks and fingerprints. A watermark is a message that allows ownership of the marked content to be proven, whereas a fingerprint allows buyer identification.

Collusion attacks are not an issue for watermarking (all marked copies being identical), but should be considered in the case of fingerprinting. In a collusion attack, a set of dishonest buyers compare their copies in order to locate differences between them and try to fabricate a new content whose mark is either no longer recoverable or does not allow identification of any of the colluders.
In [1,2], the concept of fingerprinting secure against buyer collusions is introduced. A general construction is given to obtain fingerprinting codes secure against collusions of up to c buyers (c-secure codes). For N possible buyers and given $\epsilon > 0$, $L = 2c\log(2N/\epsilon)$ and $d = 8c^2\log(8cL/\epsilon)$ a code with N codewords of length

$$l = 2Ldc = 32c^4\log(2N/\epsilon)\log(8cL/\epsilon) \quad (1)$$

* This work has been partly supported by the European Commission under project IST-2001-32012 "Co-Orthogonal Codes" and by the Spanish Ministry of Science and Technology and the European FEDER fund through project no. TIC2001-0633-C03-01 "STREAMOBILE".

L. Batten and J. Seberry (Eds.): ACISP 2002, LNCS 2384, pp. 316–327, 2002.

is constructed which allows one of the colluders to be identified with probability $1-\epsilon$. The authors also show that, for $c \geq 2$ and $N \geq 3$, it is not possible to obtain c-secure codes where colluders are identified with probability 1.
In [3] it is shown that, for $c = 2$, collusion security can be obtained using the error-correcting capacity of dual Hamming codes. In this way, 2-secure fingerprinting codes are obtained which are much shorter than 2-secure codes obtained via the general construction [1,2].
We show in this paper that, for $c = 3$, it is also possible to come up with collusion-secure fingerprinting codes much shorter than 3-secure codes obtained from the general construction [1,2]. The basic idea is to compose a new kind of code, which we call *scattering code*, with a dual Hamming code.
Section 2 presents some results on dual Hamming codes. Section 3 presents a set of lemmas on the probability of successful collusion as a function of the strategy of colluders. The construction and decoding of scattering codes are introduced in Section 4. Finally, Section 5 explains how to generate fingerprinting codes secure against collusions of up to three buyers by composing a dual Hamming code with a scattering code. Section 6 compares the length of our proposal codewords and those of [1,2]. Section 7 is a conclusion. *The Appendix contains proofs for all presented results.*

2 Dual Binary Hamming Codes

The dual code of a binary Hamming code (denoted by $DH(n)$) is a binary code with 2^n codewords of length $N = 2^n - 1$ such that the distance between any two codewords is 2^{n-1} (see [4] for an introduction). A few definitions and properties related to such codes are presented next.

Definition 1. Let a^1, a^2, a^3 be three codewords of a $DH(n)$ code, *i.e.* $a^i = a^i_1 a^i_2 \cdots a^i_N$. Define $inv(a^1, a^2, a^3)$ as the set of invariant positions between all three codewords, that is, those bit positions in which all three codewords have the same bit value. Formally speaking,

$$inv(a^1, a^2, a^3) = \{i,\ 1 \leq i \leq N,\ a^1_i = a^2_i = a^3_i\}$$

Definition 2. Let a^1, a^2, a^3 be three codewords of a $DH(n)$ code. Define $minor(a^1; a^2, a^3)$ as the set of bit positions in which a^1 has a value different from the values in a^2 and a^3 (for such positions, $a^2_i = a^3_i$). Formally speaking,

$$minor(a^1; a^2, a^3) = \{i,\ 1 \leq i \leq N,\ a^1_i \neq a^2_i,\ a^1_i \neq a^3_i\}$$

Lemma 3. *Let a^1, a^2, a^3 be three codewords of a $DH(n)$ code. Then it holds that $|inv(a^1, a^2, a^3)| = 2^{n-2} - 1$, $|minor(a^1; a^2, a^3)| = 2^{n-2}$, $|minor(a^2; a^1, a^3)| = 2^{n-2}$ and $|minor(a^3; a^1, a^2)| = 2^{n-2}$.*

Lemma 4. *Let a^1, a^2, a^3 be three codewords of a $DH(n)$ code. Then it holds that:*

- *There exists one and only one codeword $a^z \in DH(n) \backslash \{a^1, a^2, a^3\}$ such that $a_i^z = a_i^1 = a_i^2 = a_i^3$, $\forall i \in inv(a^1, a^2, a^3)$. Furthermore, $a_i^z = a_i^1$, $\forall i \in minor(a^1; a^2, a^3)$, $a_i^z = a_i^2$, $\forall i \in minor(a^2; a^1, a^3)$ and $a_i^z = a_i^3$, $\forall i \in minor(a^3; a^1, a^2)$.*
- *The remaining codewords satisfy that $\forall a^j \in DH(n) \backslash \{a^1, a^2, a^3, a^z\}$, $d_{inv(a^1,a^2,a^3)}(a^j, a^1) = d_{minor(a^1;a^2,a^3)}(a^j, a^1) = d_{minor(a^2;a^1,a^3)}(a^j, a^1) = d_{minor(a^3,a^1,a^2)}(a^j, a^1) = 2^{n-3}$, where $d_P(x, y)$ denotes Hamming distance between codewords x and y restricted to bit positions in P. The same distances hold with respect to a^2 and a^3.*

3 3-Collusions over $DH(n)$

Let us assume that three dishonest buyers c^1, c^2, c^3 compare their copies of the same multimedia content. According to the marking assumption [1,2], they can only modify the embedded marks in those *detectable* positions where not all three marks take the same bit value. In those positions, the colluders can set the corresponding bit to 0, 1 or "unreadable". In this way, we conclude that, if three different buyers are assigned codewords a^1, a^2 and a^3 of a $DH(n)$ code, the result of their collusion will be a codeword a^{coll} where no bit has been modified in the $2^{n-2} - 1$ positions in $inv(a^1, a^2, a^3)$. On the other hand, colluders will be able to identify positions in $minor(a^1; a^2, a^3)$ as the bit positions of those content fragments which are identical between the copies of c^2 and c^3 and different from the copy of c^1. In a similar way, $minor(a^2; a^1, a^3)$ and $minor(a^3; a^1, a^2)$ can be identified as well.

In order for a collusion to be successful, colluders must generate, by mixing fragments in their copies, a codeword such that the closest codeword in the fingerprinting code is not in $\{a^1, a^2, a^3\}$. In this way, another buyer will be accused in lieu of the colluders. Intuitively, it can be realized that a reasonable strategy to come up with a codeword as distant as possible from the colluders' codewords is to build that codeword in such a way that each colluder contributes the same number of bits.

Definition 5. A *p-majority* collusion strategy is one in which colluders choose with probability p the majority bit value in positions $minor(a^i; a^j, a^k)$ (that is, the bit values in a^j or a^k).

Lemma 6. *Let a^{coll} be a codeword that has been generated using a p-majority collusion strategy between three codewords $a^1, a^2, a^3 \in DH(n)$. It holds that $d_1 = d(a^{coll}, a^i) = K_1, \forall i = 1, 2, 3$ with*

$$p_1(k) = p(K_1 = k) = \sum_{t=\max(0,k-2^{n-1})}^{\min(k,2^{n-2})} b(t; 2^{n-2}, p) b(k - t; 2^{n-1}, 1 - p)$$

where $b(x_1; x_2, x_3) = \binom{x_2}{x_1} x_3^{x_1}(1-x_3)^{x_2-x_1}$ *is the binomial probability function (*x_2 *is the number of trials,* x_3 *the success probability per trial and* x_1 *is the number of successful trials).*

Lemma 7. *Let* a^{coll} *be a codeword generated using a p-majority collusion strategy between three codewords* $a^1, a^2, a^3 \in DH(n)$*. It holds that* $d_2 = \min_{i=1,2,3} d(a^{coll}, a^i) = K_2$ *with*

$$p_2 = p(K_2 = k) = \sum_{i=1}^{3} \binom{3}{i} p_1(k)^i \Big[\sum_{k'>k} p_1(k') \Big]^{3-i}$$

Lemma 8. *Let* a^{coll} *be a codeword generated using a p-majority strategy between three codewords* $a^1, a^2, a^3 \in DH(n)$ *and let* a^z *be the only codeword in* $DH(n)\backslash\{a^1, a^2, a^3\}$ *with* $a_i^z = a_i^1 = a_i^2 = a_i^3$, $\forall i \in inv(a^1, a^2, a^3)$ *(existence and uniqueness of* a^z *are guaranteed by Lemma 4). Then,* $d_3 = d(a^z, a^{coll}) = K_3$ *with*

$$p_3(k) = p(K_3 = k) = b(k; 3 \cdot 2^{n-2}, p)$$

Lemma 9. *Let* a^{coll} *be a codeword generated using a p-majority strategy between three codewords* $a^1, a^2, a^3 \in DH(n)$ *and let* a^z *be the only codeword in* $DH(n)\backslash\{a^1, a^2, a^3\}$ *with* $a_i^z = a_i^1 = a_i^2 = a_i^3$, $\forall i \in inv(a^1, a^2, a^3)$*. Then, for any codeword* $a \in DH(n)\backslash\{a^1, a^2, a^3, a^z\}$*, it holds that* $d_4 = d(a, a^{coll}) = 2^{n-3} + K_4$ *with*

$$p_4(k) = p(K_4 = k) = \sum_{t=\max(0,k-3\cdot 2^{n-3})}^{\min\{k,3\cdot 2^{n-3}\}} b(t; 3 \cdot 2^{n-3}, 1-p) b(k-t; 3 \cdot 2^{n-3}, p)$$

For the sake of simplicity, let us assume in what follows that d_3 is distributed like d_4. Since for $p > 0.\hat{6}$ the number of different bits expected for d_3 is greater than the number of different bits expected for d_4, such a distributional assumption will cause actual security to be even slightly higher than computed in what follows.

Lemma 10. *Let* a^{coll} *be a codeword generated using a p-majority strategy between three codewords* $a^1, a^2, a^3 \in DH(n)$*. It holds that* $d_5 = \min_{i \notin \{1,2,3\}} \{d(a^{coll}, a^i)\} = 2^{n-3} + K_5$*, with*

$$p_5(k) = p(K_5 = k) = \sum_{i=1}^{2^n-3} \binom{2^n-3}{i} p_4(k)^i \Big[\sum_{k'>k} p_4(k') \Big]^{2^n-3-i}$$

Lemma 11. *Let a^{coll} be a codeword generated using a p-majority strategy between three codewords $a^1, a^2, a^3 \in DH(n)$. The probability that the codeword in $DH(n)$ closest to a^{coll} is* not *in $\{a^1, a^2, a^3\}$ is expressed by*

$$\epsilon = \sum_{k=0}^{2^n-1} p(d_2 = k)p(d_5 \leq k)$$

ϵ is the probability that decoding a^{coll} yields as a result a codeword different from any of the colluders' codewords, that is, the probability of an honest buyer being unjustly accused instead of the colluders.

Table 1. Probability ϵ of success of a 3-collusion in $DH(7)$ and $DH(8)$ for several values of p

	p					
	0.0	$0.\hat{6}$	0.7	0.8	0.9	1.0
$DH(7)$	1.0	$0.59 \cdot 10^{-3}$	$0.14 \cdot 10^{-3}$	$0.14 \cdot 10^{-6}$	$0.77 \cdot 10^{-14}$	0.0
$DH(8)$	1.0	$0.17 \cdot 10^{-7}$	$0.10 \cdot 10^{-7}$	$0.15 \cdot 10^{-13}$	$0.70 \cdot 10^{-28}$	0.0

It can be observed from Table 1 that, as n increases and p approaches 1, the probability $1 - \epsilon$ of correctly identifying one of the colluders can be made arbitrarily close to 1. *The problem is that the parameter p defining the collusion strategy is chosen by the colluders, which implies they can take $p = 0$ to make sure they are not identified!* In Section 4, we present a technique to prevent colluders from avoiding identification in this way.

4 Scattering Codes

4.1 Construction

We define a *scattering code* $SC(d, t)$ with parameters (d, t) as a binary code consisting of $2t$ codewords of length $(2t + 1)d$ constructed as follows:

1. Start generating the codewords of $SC(1, t)$ as:
   a) The i-th codeword for $1 \leq i \leq t$ is constructed by setting the first and the $(i + 1)$-th bits of the codeword to '1'. The remaining bits are set to '0'.
   b) The i-th codeword for $t + 1 \leq i \leq 2t$ is constructed by setting the $(i + 1)$-th bit of the codeword to '1'. The remaining bits are set to '0'.
2. The code $SC(d, t)$ is generated by replicating d times every bit in the codewords of $SC(1, t)$. Define a *block* to be a group of d replicated bits.
3. By convention, the first t codewords of $SC(d, t)$ are defined to encode a '1' and the last t codewords are defined to encode a '0'. The first block of the code is called 'Zone-A', the next t blocks are called 'Zone-B' and the last t blocks are called 'Zone-C'. See an example in Table 2.

Table 2. Codewords of a scattering code $SC(4, 3)$

Encodes	Zone-A	Zone-B	Zone-C
'1'	1111	1111 0000 0000	0000 0000 0000
	1111	0000 1111 0000	0000 0000 0000
	1111	0000 0000 1111	0000 0000 0000
'0'	0000	0000 0000 0000	1111 0000 0000
	0000	0000 0000 0000	0000 1111 0000
	0000	0000 0000 0000	0000 0000 1111

Using scattering codes, a '1' is encoded by randomly choosing one of the first t codewords and a '0' is encoded by randomly choosing one of the last t codewords.

4.2 Decoding

A scattering code is decoded by using the first applicable rule among the following ordered list:

1. If all bits in 'Zone-A' are '1' and all bits in 'Zone-C' are '0', decode as '1'.
2. If all bits in 'Zone-A' are '0' and all bits in 'Zone-B' are '0', decode as '0'.
3. If in two blocks of 'Zone-B' there is at least one bit with value '1' in each one, decode as '1'.
4. If in two blocks of 'Zone-C' there is at least one bit with value '1' in each one, decode as '0'.
5. If there are more '1' bits than '0' bits in 'Zone-A', decode as '1'.
6. If there are more '0' bits than '1' bits in 'Zone-A', decode as '0'.
7. Decode as 'Unreadable'

Lemma 12. *Let b^{coll} be a codeword generated by using a p-majority strategy between three codewords $b^1, b^2, b^3 \in SC(d, t)$ encoding the same bit value v. Then, b^{coll} decodes as v with probability* 1.

Lemma 13. *Let b^{coll} be a codeword generated using a p-majority strategy between three codewords $b^1, b^2, b^3 \in SC(d, t)$, with two of them ($b^1$ and b^2) encoding a value v and the other (b^3) a value $\overline{v}$. Then, the probability that b^{coll} decodes as v is given by*

$$p(v) = (1 - \frac{1}{t})p_{diff}(v) + \frac{1}{t}p_{same}(v)$$

where $p_{diff}(v)$ is the probability of decoding as v when $b^1 \neq b^2$ and can be computed as $p_{diff}(v) = 1 - p_{diff}(\overline{v})$ and

$$\begin{aligned} p_{diff}(\overline{v}) = {} & (1-p)^d p^{2d} + \\ & + 2 \cdot p^d (1-p^d) \textstyle\sum_{k=0}^{\lfloor \frac{d-1}{2} \rfloor} b(k; d, p) + \\ & + p^{2d} \textstyle\sum_{k=1}^{\lfloor \frac{d-1}{2} \rfloor} b(k; d, p) \end{aligned}$$

and $p_{same}(v)$ *is the probability of decoding as* v *when* $b^1 = b^2$ *and can be computed as*

$$\begin{aligned} p_{same}(v) = & p^{2d} + \\ & + (1-p^d) \sum_{k=\lfloor \frac{d+2}{2} \rfloor}^{d} b(k;d,p) + \\ & + p^d \sum_{k=\lfloor \frac{d+2}{2} \rfloor}^{d-1} b(k;d,p) + \end{aligned}$$

Table 3 shows, for several codes $SC(d,t)$, the least probability $p(v)$ of decoding as the majority bit in a collusion of three codewords (two encoding v and one $\overline{v}$).

Table 3. Probability $p(v)$ of decoding as the majority bit v in a collusion of three buyers, for several parameter choices (d,t).

d	t	$\min p(v)$
3	4	0.68
5	5	0.8
7	9	0.89
31	100	0.99

5 Construction of 3-Secure Code

If there are $N = 2^n - 1$ buyers, each buyer c^i is assigned a different codeword $a^i \in DH(n)$. Rather than directly embedding a^i in the content to be sold, the merchant generates a codeword A^i by composing a scattering code $SC(d,t)$ with a^i. Such a composition is performed by replacing each bit of a^i with a codeword in $SC(d,t)$ that encodes the value of the bit of a^i. In this way, the codeword A^i will have bitlength

$$l = (N-1)(2t+1)d \tag{2}$$

The merchant then permutes the bits in A^i using a pseudo-random permutation seeded by a secret key known only to the merchant. Finally, the merchant embeds the permuted version of A^i in the content being sold.

What is achieved with the above composition is that, regardless of the p'-majority strategy used by colluders to generate codewords A^{coll}, the $p(v)$-majority strategy resulting from decoding A^{coll} as a^{coll} has a value $p(v)$ that can be controlled by the merchant by choosing appropriate values for parameters d and t (see Table 3). It can be seen from Table 1 that controlling $p(v)$ is necessary to the keep low the probability ϵ of successful collusion. If A^i has some bits with value 'Unreadable', those bits are randomly set to '0' or '1'.

6 Numerical Results

Once parameters d and t have been chosen, the number of buyers can be increased by increasing n. Table 4 shows, for $d = 5$ and $t = 5$, the size of the code (number of buyers), the length of codewords in our proposal, the probability ϵ of a successful collusion and the length of codewords in the Boneh-Shaw proposal for the same parameters. It can be seen that the Boneh-Shaw construction requires much longer codewords than our proposal. Furthermore, as n increases, their codeword length increases faster than ours.

Table 4. Codeword length comparison between our proposal and Boneh-Shaw's for the same number of users n and security level ϵ (parameter choice $d = 5$, $t = 5$)

n	buyers	ϵ	Our length	Boneh-Shaw's length
7	128	$0.14 \cdot 10^{-6}$	6,985	2,788,320
8	256	$0.15 \cdot 10^{-13}$	14,025	8,393,220
9	512	$0.19 \cdot 10^{-27}$	28,105	28,340,928

In our proposal, once d and t have been fixed, the value ϵ decreases exponentially as n increases, so that the security level reached may be unnecessarily high. A better comparison is to use a fixed ϵ assuming that, for our security requirements, any $\epsilon' < \epsilon$ can be regarded as negligible. Table 5 presents a codeword length comparison for $\epsilon = 10^{-10}$. It can be observed that, for $\epsilon = 10^{-10}$, our proposal is shorter up to $n = 16$ (number of buyers below 65,536). For values of $n > 16$, Boneh-Shaw's proposal has a shorter codeword length. This is due to the fact that our codeword length increases as $O(N)$ while Boneh-Shaw's increases as $O(\log N)$. Nonetheless, the Boneh-Shaw proposal would only be *substantially* shorter than ours if the number of buyers were *really huge*; for usual figures, our proposal is the one substantially shorter.

Table 5. Codeword length comparison between our proposal and Boneh-Shaw's assuming $\epsilon = 10^{-10}$

buyers	Our length	Boneh-Shaw's length
512	28,105	5,148,000
1,024	56,265	5,269,992
...	...	...
32,768	1,802,185	5,883,888
65,536	3,604,425	6,006,780
131,072	7,208,905	6,129,816

7 Conclusion

Codes for fingerprinting have been presented which, for collusions of up to three buyers and not too large a number of possible buyers, require a codeword length much shorter than the one required by the Boneh-Shaw construction. Thus, the proposed scheme is especially useful when the capacity of the embedding scheme is low, as it may happen when protecting precision-critical content (medical images, etc.). On the other hand, collusions tend to be small due to their clandestine nature (joining a large group of colluders is risky for a buyer); therefore, there is plenty of use for codes that, like the ones presented here, provide efficient protection against small collusions.

References

1. D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data", in *Advances in Cryptology-CRYPTO'95*, LNCS 963, pp. 452-465, 1995.
2. D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data", *IEEE Trans. Inf. Theory*, vol **IT-44**, no. (5), pp. 1897-1905, 1998.
3. J. Domingo-Ferrer and J. Herrera-Joancomartí, "Short collusion-secure fingerprints based on dual binary Hamming codes", *Electronics Letters*, vol. 36, no. 20, pp. 1697-1699, 2000.
4. F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam: North-Holland, 1977.

Appendix

Proof (Lemma 3): Let a^1, a^2, a^3 be three codewords of a $DH(n)$ code. Since $d(a^i, a^j)_{i \neq j} = 2^{n-1}$, then $|inv(a^1, a^2)| = 2^{n-1} - 1$. Let $x = |inv(a^1, a^2, a^3)|$; then $|minor(a^3; a^1, a^2)| = 2^{n-1} - 1 - x$. Let $y = |minor(a^2; a^1, a^3)|$; then $|minor(a^1; a^3, a^2)| = 2^{n-1} - y$. On the other hand, $|inv(a^1, a^3)| = x + y = 2^{n-1} - 1$ and $|inv(a^2, a^3)| = 2^{n-1} - 1 = x + 2^{n-1} - y$. By solving the following equation system,

$$\begin{cases} x + y = 2^{n-1} - 1 \\ 2^{n-1} - 1 = x + 2^{n-1} - y \end{cases}$$

we get $x = 2^{n-2} - 1$ and $y = 2^{n-2}$ and thus $|inv(a^1, a^2, a^3)| = 2^{n-2} - 1$, $|minor(a^1; a^3, a^2)| = 2^{n-2}$, $|minor(a^2; a^1, a^3)| = 2^{n-2}$ and $|minor(a^3; a^1, a^2)| = 2^{n-2}$.

Proof (Lemma 4): First of all, we prove the existence and properties of a^z. As a $DH(n)$ code is a linear code, any linear combination of codewords results in another codeword. Then, we get $a^z = a^1 \oplus a^2 \oplus a^3$, where $\oplus$ denotes the component-wise modulo 2 addition.

We prove that $a^z_i = a^1_i = a^2_i = a^3_i$, $\forall i \in inv(a^1, a^2, a^3)$. This is true because if $a^1_i = a^2_i = a^3_i = 1$, then $a^1_i \oplus a^2_i \oplus a^3_i = 1$, and if $a^1_i = a^2_i = a^3_i = 0$, then $a^1_i \oplus a^2_i \oplus a^3_i = 0$.

Then, we prove $a_i^z = a_i^1,\ \forall i \in minor(a^1; a^2, a^3)$. This is true because $a_i^z = a_i^1 \oplus a_i^2 \oplus a_i^3$ and as $a_i^2 = a_i^3$, then $a_i^z = a_i^1$.

Using the same procedure we can prove $a_i^z = a_i^2,\ \forall i \in minor(a^2; a^1, a^3)$ and $a_i^z = a_i^3,\ \forall i \in minor(a^3; a^1, a^2)$.

Next we prove the second part of the Lemma. Let $a^j \in DH(n)\backslash\{a^1, a^2, a^3, a^z\}$
Call x the number of positions in $inv(a^1, a^2, a^3)$ where $a_i^j = a_i^1$. Then the number of positions in $inv(a^1, a^2, a^3)$ where $a_i^j \neq a_i^1$ is $2^{n-2} - 1 - x$ (see Lemma 3).
Call y the number of positions in $minor(a^1; a^2, a^3)$ where $a_i^j = a_i^1$. Then the number of positions in $minor(a^1; a^2, a^3)$ where $a_i^j \neq a_i^1$ is $2^{n-2} - y$.
Call z the number of positions in $minor(a^2; a^1, a^3)$ where $a_i^j = a_i^1$. Then the number of positions in $minor(a^2; a^1, a^3)$ where $a_i^j \neq a_i^1$ is $2^{n-2} - z$.
Call t the number of positions in $minor(a^3; a^1, a^2)$ where $a_i^j = a_i^1$. Then the number of positions in $minor(a^3; a^1, a^2)$ where $a_i^j \neq a_i^1$ is $2^{n-2} - t$.
As $d(a^j, a^1) = d_{inv(a^1,a^2,a^3)}(a^j, a^1) + d_{minor(a^1;a^2,a^3)}(a^j, a^1) + d_{minor(a^2;a^1,a^3)}(a^j, a^1) + d_{minor(a^3,a^1,a^2)}(a^j, a^1) = 2^{n-1}$, then

$$(2^{n-2} - 1 - x) + (2^{n-2} - y) + (2^{n-2} - z) + (2^{n-2} - t) = 2^{n-1}$$

As $d(a^j, a^2) = d_{inv(a^1,a^2,a^3)}(a^j, a^2) + d_{minor(a^1;a^2,a^3)}(a^j, a^2) + d_{minor(a^2;a^1,a^3)}(a^j, a^2) + d_{minor(a^3,a^1,a^2)}(a^j, a^2) = 2^{n-1}$, then

$$(2^{n-2} - 1 - x) + (2^{n-2} - y) + z + t = 2^{n-1}$$

As $d(a^j, a^3) = d_{inv(a^1,a^2,a^3)}(a^j, a^3) + d_{minor(a^1;a^2,a^3)}(a^j, a^3) + d_{minor(a^2;a^1,a^3)}(a^j, a^3) + d_{minor(a^3,a^1,a^2)}(a^j, a^3) = 2^{n-1}$, then

$$(2^{n-2} - 1 - x) + y + (2^{n-2} - z) + t = 2^{n-1}$$

As $d(a^j, a^z) = d_{inv(a^1,a^2,a^3)}(a^j, a^z) + d_{minor(a^1;a^2,a^3)}(a^j, a^z) + d_{minor(a^2;a^1,a^3)}(a^j, a^z) + d_{minor(a^3,a^1,a^2)}(a^j, a^z) = 2^{n-1}$, then

$$(2^{n-2} - 1 - x) + y + z + (2^{n-2} - t) = 2^{n-1}$$

From the expressions above we build the following equation system:

$$\begin{cases} x + y + z + t = 2^{n-1} - 1 \\ -x - y + z + t = 1 \\ -x + y - z + t = 1 \\ -x + y + z - t = 1 \end{cases}$$

By solving it, we get $x = 2^{n-3} - 1$ and $y = z = t = 2^{n-3}$.
Finally, we conclude,
$d_{inv(a^1,a^2,a^3)}(a^j, a^1) = 2^{n-2} - 1 - x = 2^{n-3}$
$d_{minor(a^1;a^2,a^3)}(a^j, a^1) = 2^{n-2} - y = 2^{n-3}$
$d_{minor(a^2;a^1,a^3)}(a^j, a^1) = 2^{n-2} - z = 2^{n-3}$
$d_{minor(a^3;a^1,a^2)}(a^j, a^1) = 2^{n-2} - t = 2^{n-3}$

In the same way, we can prove that these distances hold between a^j and a^2, a^3.

Proof (Lemma 6): Without loss of generality, take $i = 1$. We have that, for bit positions in $inv(a^1, a^2, a^3)$, there is no difference between a^1 and a^{coll} since bits in those positions are undetectable. Also, each of the 2^{n-2} bits in $minor(a^1; a^2, a^3)$ differs between a^1 and a^{coll} with probability p; therefore, the probability of there being t differing bits in those positions is given by a binomial probability function $b(t; 2^{n-2}, p)$. Also, each of the $2 \cdot 2^{n-2}$ bits in $minor(a^2; a^1, a^3)$ and $minor(a^3; a^1, a^2)$ differs between a^1 and a^{coll} with probability $(1-p)$; therefore, the probability of there being $k-t$ differing bits in those positions is given by a binomial probability function $b(k-t; 2^{n-1}, 1-p)$. In this way, the expression in the Lemma corresponds to the probability of there being a total of $t + (k-t) = k$ differing bits between a^1 and a^{coll}.

Proof (Lemma 7): The expression in the Lemma corresponds to the probability of one, two or three codewords in $\{a^1, a^2, a^3\}$ being at distance k from a^{coll} and the remaining codewords being at a greater distance.

Proof (Lemma 8): Lemma 4 says that bits of a^z are identical to bits of a^i in the positions in $minor(a^i; a^j, a^k)$ for $(i, j, k) \in \{(1, 2, 3), (2, 1, 3), (3, 1, 2)\}$. Therefore, the probability of there being k different bits in those $3 \cdot 2^{n-2}$ positions is given by a binomial probability function $b(k; 3 \cdot 2^{n-2}, p)$.

Proof (Lemma 9): According to Lemma 4, a^{coll} and a have 2^{n-3} differing bits in positions in $inv(a^1, a^2, a^3)$. In each $minor(a^i; a^j, a^k)$, for $(i, j, k) \in \{(1, 2, 3), (2, 1, 3), (3, 1, 2)\}$, a^{coll} has all 2^{n-3} bits each of which is different with probability p and 2^{n-3} bits each of which is different with probability $(1-p)$. Therefore, we have $3 \cdot 2^{n-3}$ bits with probability p of being different, and thus the probability that t of such bits are different is $b(t; 3 \cdot 2^{n-3}, p)$. On the other hand, we have $3 \cdot 2^{n-3}$ bits with probability $1-p$ of being different, and thus the probability that $k-t$ of such bits are different is $b(k-t; 3 \cdot 2^{n-3}, 1-p)$. In this way, the expression in the Lemma computes the probability of there being $t + (k-t) = k$ different bits between a and a^{coll}.

Proof (Lemma 10): The expression in the Lemma computes the probability that at least one out of the $2^n - 3$ codewords in $DH(n) \backslash \{a^1, a^2, a^3\}$ is at distance k of a^{coll}, with the remaining codewords at a greater distance.

Proof (Lemma 12): It can be seen that, if $v = 1$, bits in 'Zone-A' and in 'Zone-C' stay undetectable and thus decoding will use Rule 1 and return a value 1.

If $v = 0$, bits in 'Zone-A' and in 'Zone-B' stay undetectable. Thus, decoding will use Rule 2 and return a value 0.

Proof (Lemma 13): In a collusion between three codewords $b^1, b^2, b^3 \in SC(d,t)$ with two of them (b^1 and b^2) encoding a value v (without loss of generality, assume $v = 1$ and $\overline{v} = 0$), we have $b^1 = b^2$ with probability $\frac{1}{t}$ and $b^1 \neq b^2$ with probability $1 - \frac{1}{t}$.

In the case $b^1 \neq b^2$, we compute $p_{\text{diff}}(v) = 1 - p_{\text{diff}}(\overline{v})$ (it can be shown that for odd values of d Rule 7 is never reached), where $p_{\text{diff}}(\overline{v})$ corresponds to the probability of decoding $\overline{v}$ after a collusion based on a p-majority strategy. $p_{\text{diff}}(\overline{v})$ is actually the probability of decoding using Rules 2 or 6 (Rule 4 is never used when $b^1 \neq b^2$).

In the case $b^1 = b^2$, we compute $p_{\text{same}}(v)$ as the probability of decoding v after a collusion based on a p-majority strategy. $p_{\text{same}}(v)$ is actually the probability of decoding using Rules 1 or 5 (Rule 3 is never used when $b^1 = b^2$).