On the Leakage of Information in Biometric Authentication

Elena Pagnin, Christos Dimitrakakis, Aysajan Abidin, Aikaterini Mitrokotsa

Chalmers University of Technology, Gothenburg, Sweden
{elenap, chrdimi, aysajan.abidin, aikmitr}@chalmers.se

Abstract. In biometric authentication protocols, a user is authenticated or granted access to a service if her fresh biometric trait *matches* the reference biometric template stored on the service provider. This matching process is usually based on a suitable *distance* which measures the similarities between the two biometric templates. In this paper, we prove that, when the matching process is performed using a specific family of distances (which includes distances such as the Hamming and the Euclidean distance), then information about the reference template is leaked. This leakage of information enables a *hill-climbing* attack that, given a sample that matches the template, could lead to the full recovery of the biometric template (*i.e.* centre search attack) even if it is stored encrypted. We formalise this "leakage of information" in a mathematical framework and we prove that centre search attacks are feasible for any biometric template defined in $\mathbb{Z}_q^n, (q \geqslant 2)$ after a number of authentication attempts linear in n. Furthermore, we investigate brute force attacks to find a biometric template that matches a reference template, and hence can be used to run a *centre search attack*. We do this in the binary case and identify connections with the *set-covering* problem and *sampling without replacement*.

Key words: Biometric authentication, privacy-preservation, centre search attack, hill-climbing, brute force attacks.

1 Introduction

While biometric authentication is becoming increasingly popular, the privacy and security risks related to their usage are raising severe concerns. The main threats associated to biometric authentication include profiling and tracking of individuals and identity theft. If successfully performed, any attack that recovers a biometric template may have serious impact since users cannot change their biometric features and biometric data may reveal very sensitive information (*e.g.* genetic [1] information and medical diseases [2]).

Biometric authentication protocols involve comparing fresh biometric data with a stored biometric template. The process is essentially performed by computing some distance or divergence between the fresh and the stored template. If the measured distance is less than a predefined threshold, then the user is authenticated; otherwise she is rejected. Many biometric authentication protocols use straightforward choices for the distance, such as the Hamming distance [3, 4], the normalised Hamming distance ([5] for iris recognition) and the Euclidean distance [6–9]. In these cases the matching process leaks information that could be exploited by an adversary to recover the stored template. More precisely, the adversary could run an iterative process where he progressively changes the components of an arbitrary biometric template until acceptance. This strategy is known as *hill-climbing* attack [10], due to similarity with the synonymous optimisation technique. When the initial template is an acceptable biometric trait (*e.g.* a fresh sample) this process is called *centre search* attack [10]. Recovering stored biometric templates has more severe impact than just finding an acceptable biometric template. Indeed, the same stored template might be used in multiple biometric authentication systems which may even employ different matching processes. Furthermore, a recovered stored template could be used to find a match in criminal biometric template databases or even compromise health records [11].

Bringer *et al.* [12] presented a hill-climbing strategy that is successful even when a dedicated secure access module (*e.g.* smartcard) is used to perform the biometric authentication process. The matching process considered in [12] involves an adapted Hamming distance with erasures, nevertheless, the adversary is able to recover multiple encrypted biometric templates. Later on, Simoens *et al.* [10] describe multiple attacks (including the centre search attack) that can be mounted by each of the internal entities in a distributed biometric authentication systems.

In the past years privacy-preserving distance computation has been investigated [13–15]. Although these protocols have direct applications to biometric identification and authentication they all suffer from leakage of information when a centre search attack is employed.

The problem of leakage of information due to the employment of distances has also been investigated in other areas not relevant to biometric authentication. For example, the Hamming weight model has been employed in order to successfully perform side channel attacks [16, 17] (*e.g.* differential power analysis). It has been shown [16, 17] that the power consumption of a device (*e.g.* a smart card) directly depends on the Hamming

weight and on the number of changes $0 \leftrightarrow 1$ in the binary vector that is considered during the execution of the attack.

Our contribution: In this paper, we point out that all biometric authentication protocols that rely on certain distances (including the Hamming and the Euclidean distance) are susceptible to leakage of information and we provide a formal mathematical framework to analyse this. In particular, we generalise the centre search attack and prove that it is efficient and feasible in the binary case as well as when the biometric templates are defined in $\mathbb{Z}_q^n$. In both cases we show that the maximal number of authentication attempts in order to fully recover the stored biometrics corresponding to the given data is linear in n (the size of the biometric string). Our proofs hold also when the Euclidean distance is employed. Thus, we go beyond the Hamming distance case that was described in [10]. We furthermore investigate the preliminary step to the centre search attack: finding a biometric template that matches a reference one. For the binary case, we propose a new algorithm that exploits a tree structure and we compare its performance to standard brute force attacks and to the *optimal* but infeasible attack. Finally, we highlight how the *optimal* solution of finding a matching biometric template connects to the NP-complete *set-covering* problem and *sampling without replacement*. Our proofs are valid for standard as well as for privacy-preserving biometric authentication protocols since the output of the matching process is not affected by the employed protection mechanism (*e.g.* homomorphic encryption). This means that encryption alone cannot mitigate the leakage of information of the matching process. More precisely, this leakage of information leads to full recovery of the stored template for the centre search attack and to a matching template for the brute-force attack. An implication of our work is that achieving security and privacy of biometric templates using the known techniques is challenging.

Outline: The notations and the background material are introduced in Section 2 while Section 3 describes the adversarial model. We generalise the centre search attack in Section 4 in two ways: first to any leaking distance on $\mathbb{Z}_2^n$ and then to any leaking distance on $\mathbb{Z}_q^n$. In addition, we investigate the success probability of finding an acceptable fresh biometric template and compare the bounds for the success probability in different cases in Section 5. Finally, Section 6 summarizes our results.

2 Preliminaries

Notations: Let $q \in \mathbb{Z}$ be a positive integer, $q \geqslant 2$. The set of n-dimensional vectors with components in $\mathbb{Z}_q = \{0, 1, \cdots, q-1\}$ is denoted by $\mathbb{Z}_q^n$. The i-th component of a vector $x \in \mathbb{Z}_q^n$ is referred to as $x_i \in \mathbb{Z}_q$. Given a distance $d : \mathbb{Z}_q^n \times \mathbb{Z}_q^n \to \mathbb{R}_{\geqslant 0}$, a point $x \in \mathbb{Z}_q^n$ and a positive number $\tau \in \mathbb{R}_{>0}$, the d-ball of center x and radius τ is defined as $B_x(\tau) = \{z \in \mathbb{Z}_q^n \ : \ d(x, z) \leqslant \tau\}$. In the following, the binary case ($q = 2$) will always be explicitly written as $\mathbb{Z}_2^n$. If not otherwise specified, $\mathbb{Z}_q^n$ implies $q > 2$. We denote the bit-flip operation as $\bar{\ }: \mathbb{Z}_2 \to \mathbb{Z}_2$, namely $\bar{1} = 0$, $\bar{0} = 1$. The integer part of a real number τ, is denoted by $\lfloor \tau \rceil$ (rounding to the closest integer $\leqslant \tau$).

2.1 Biometric authentication

A biometric authentication system consists of two main phases: the *enrolment phase* and the *authentication phase.*

The *enrolment phase* is a one-time step: a user (client) $\mathcal{C}$ registers to a trusted party her biometric templates (digital strings b) along with her identity ID. These two pieces of information are then stored in the database of the authentication server $\mathcal{AS}$. Once enrolled in the system, the client can authenticate herself an unlimited number of times.

In the *authentication phase*, the client is required to provide a fresh biometric trait b' as well as her identity ID. These two data are then communicated to the authentication server, which checks if matching templates (fresh b' and stored b) match. If the distance between the user's fresh biometric trait b' and the reference biometric template b is less or equal to a predefined threshold τ, then the client gets authenticated. Otherwise, the system rejects the user.

Without loss of generality we will consider only the two party setting (*i.e.* one client $\mathcal{C}$ and one authentication server $\mathcal{AS}$, as depicted in Figure 1). However, our analysis naturally applies when more than two parties are involved in the biometric authentication process [4, 18, 19]. Due to privacy concerns, the biometric templates should be protected and not sent in the clear over the network. This implies that often the matching procedure is performed in the encrypted domain. For instance, in multiple privacy-preserving biometric authentication protocols, secure multi-party computation techniques are employed to preserve the privacy of the users. In those protocols usually the biometric data are protected

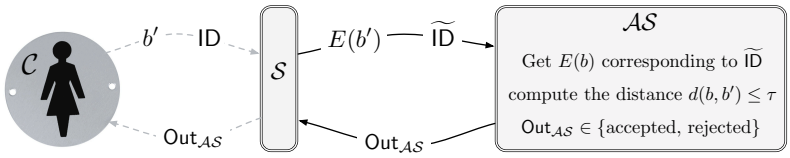

Fig. 1: Authentication phase in a two-party biometric authentication system.

using homomorphic encryption [20], garbled circuits [21] or oblivious transfer [22].

Figure 1 depicts the authentication phase of a biometric authentication system in a two party setting, between a client $\mathcal{C}$ and an authentication server $\mathcal{AS}$. The client presents her fresh biometric and her ID to the authentication system. The sensor $\mathcal{S}$ gets the user's biometric vector b' and her identity. In the privacy-preserving case, $\mathcal{S}$ encrypts b' ($E(b')$) and ID ($\widetilde{\mathsf{ID}}$), otherwise this data is sent in the clear. Subsequently, the two data ($E(b')$, $\widetilde{\mathsf{ID}}$) are sent to the authentication server $\mathcal{AS}$, who retrieves the (possibly encrypted) stored template that corresponds to the user with identity ID. The matching process is then preformed by checking if the distance between the fresh and stored biometric templates is less than a predefined threshold τ (*i.e.* $d(b, b') \leqslant \tau$). Finally, depending on the outcome of the matching ($\mathsf{Out}_{\mathcal{AS}}$), the authentication server either accepts or rejects the client. Note that even in the privacy-preserving case, where the biometric data is encrypted, the output of the authentication server depends only on the value of $d(b, b')$, *i.e.* the distance between the fresh and the stored biometric vectors. Hence, encryption alone does not mitigate our attacks.

The main enablers of the attacks described in this paper are:

(a) A return channel of the biometric authentication process, denoted as $\mathsf{Out}_{\mathcal{AS}}$ (*e.g.* access granted or not) that is sent by the authentication server to the user after each authentication attempt. In a real-life biometric authentication scenario this could be a door that opens denoting "access granted" when biometric authentication is used for access control in a building.
(b) The fact that the matching process (and so the value of $\mathsf{Out}_{\mathcal{AS}}$) is based on a distance that is sensitive to single component variations (see leaking distance Definition 1).

In this paper, we demonstrate that even when secure-multi party computation techniques are employed, it is still possible to disclose the

biometric templates as long as a certain family of distances is used to compare the raw (plaintext) biometric data. That is, an attacker can learn information about the value of b (plaintext of stored biometric template) by observing the authentication server's response $\mathsf{Out}_{\mathcal{AS}}$ to the client's authentication requests, if the response depends on the value of $d(b, b')$. More precisely, if d is a distance that detects component-variation (see Definition 1), and if there exists a function f that enables to retrieve information about the distance of the raw templates, given their possibly encrypted versions, *i.e.* $\exists f$ *s.t.* $f(E(b), E(b')) = d(b, b')$, then the biometric authentication system leaks information (in the non privacy-preserving case $E = \mathrm{id}$, is the identity map and $f = d$ is the given distance). In particular, it is always possible to disclose the original b given a matching b'. For instance, consider the case [4] where $b, b' \in \mathbb{Z}_2^n$, $d = d_H$ is the Hamming distance and E and D are the Goldwasser-Micali [23] encryption and decryption functions, respectively. Then, $d_H(b, b') = \mathsf{HW}(b \oplus b)' = \mathsf{HW}\left(D(E(b \oplus b'))\right) = \mathsf{HW}\left(D(E(b) \times E(b'))\right)$, where HW denotes the Hamming weight of a vector, *i.e.* $\mathsf{HW}(x) = \sum_{i=1}^{n} x_i$. In this case, we have $f = \mathsf{HW} \circ D \circ \times$.

3 Adversarial Model

The main threats in a privacy-preserving biometric authentication protocol are classified as follows [10]:

- *Biometric reference recovery:* the adversary tries to recover the reference (stored) biometric template b.
- *Biometric sample recovery:* the adversary tries to recover (or generate) a fresh biometric template b' that will be acceptable by the biometric authentication system.
- *Identity privacy:* the adversary tries to link a biometric template $b(i)$ of a user i to the user's identity $\mathsf{ID}(i)$.
- *Traceability and distinguishability of users:* the adversary's objective is to distinguish different users and/or trace one user in different authentication attempts.

In this paper, we focus on the two first threats only, as they apply to any biometric authentication system, privacy-preserving or not. We also consider that the adversary $\mathcal{A}$ has access to the output of the authentication process ($\mathsf{Out}_{\mathcal{AS}}$) as well as to the predefined threshold τ used in matching process. The settings for the two attacks are:

- *Biometric reference recovery:* the adversary $\mathcal{A}$ has an acceptable fresh biometric template b' at his disposal and tries to recover the stored template b (*centre search* attack).
- *Biometric sample recovery:* the adversary $\mathcal{A}$ does not have access to an acceptable fresh biometric b' but tries to find an accepted template anyway (brute force attack).

4 Generalisations of the Centre Search Attack

Let $b' \in \mathbb{Z}_q^n$ denote a fresh biometric template and $b \in \mathbb{Z}_q^n$ the reference (stored) template, for $q \geqslant 2$. The standard *centre search* attack aims at finding the point b in the centre of the *acceptance ball* $B_b(\tau) = \{\, z \in \mathbb{Z}_2^n \;:\; d(b,z) \leqslant \tau \,\}$. Simoens *et al.* [10] gave an informal description of this attack in the case d is the Hamming distance. Here, we extend this attack to a larger family of distances over $\mathbb{Z}_2^n$ (Theorem 2). In order to do so, we prove in Theorem 1 that any *leaking distance* (cf. Definition 1) over $\mathbb{Z}_2^n$ is *equivalent* to the Hamming distance. In addition, Theorem 3 proves that a centre search attack is feasible also for $b \in \mathbb{Z}_q^n$ when $q > 2$ if a *leaking distance* (e.g. the Euclidean distance) is employed in the matching process.

The family of distances we consider in this paper is defined as follows:

Definition 1 (Leaking distances). *Let $q \geqslant 2$, a distance $d : \mathbb{Z}_q^n \times \mathbb{Z}_q^n \to \mathbb{R}_{\geqslant 0}$, is said to be a leaking distance (to detect component variations) if it can be written as $d(x,y) = h\left(\sum_{i=1}^n |x_i - y_i|^k\right)$, for all $x, y \in \mathbb{Z}_2^n$, $k \in \mathbb{Q}_{>0}$ and $h : \mathbb{R} \to \mathbb{R}_{\geqslant 0}$ a monotonically strictly increasing positive function.*

The Hamming distance is an example of a leaking distance over $\mathbb{Z}_2$ (take h to be the identity map and $k = 1$). For a general $q \geqslant 2$, the Euclidean distance detects component variation (h is the square-root function and $k = 2$). Note that *leaking* distances are *reasonable* distances to be used for biometric authentication, as they enable to compare vectors (biometric data) component wise.

In order to simulate the query/access to an oracle, we introduce the following decision function.

Definition 2. *Let $q \geqslant 2$, $\tau \in \mathbb{R}_{>0}$ and let $d : \mathbb{Z}_q^n \times \mathbb{Z}_q^n \to \mathbb{R}_{\geqslant 0}$ be a distance metric. Then, for each $x \in \mathbb{Z}_q^n$, we define a decision function $\delta_x : \mathbb{Z}_q^n \to \{0,1\}$ as $\delta_x(z) = \begin{cases} 0 & \text{if } d(x,z) > \tau \\ 1 & \text{if } d(x,z) \leqslant \tau \end{cases}$.*

It is easy to see that the decision function δ_x corresponds to the output of the authentication process denoted as $\mathsf{Out}_{\mathcal{AS}}$ in Sections 2 and 3. Firstly,

we consider biometric templates as binary vectors. This is for instance the case for iris recognition based biometric authentication [5, 24]. We begin by proving that any binary leaking distance can be written in terms of the Hamming distance.

Theorem 1. *Let $d : \mathbb{Z}_2^n \times \mathbb{Z}_2^n \to \mathbb{R}_{\geqslant 0}$ be a leaking distance on $\mathbb{Z}_2^n$. Then every d-ball corresponds to a d_H-ball, with d_H being the Hamming distance.*

We provide the proof of Theorem 1 in the appendix. Observe that Theorem 1 provides a *boardwalk* among all binary leaking distances. In particular, it enables us to extend all the results concerning Hamming distance to any other leaking distance (on $\mathbb{Z}_2^n$). For example, the *correction* factor for the Euclidean distance on $\mathbb{Z}_2^n$ is $\tau = \tilde{\tau}^2$.

Theorem 2. *Let $d_H : \mathbb{Z}_2^n \times \mathbb{Z}_2^n \to \mathbb{R}_{\geqslant 0}$ be the Hamming distance and $\tau \in \mathbb{R}_{>0}$. Then, it is possible to determine the bit-values of a string x having access only to a vector $y \in B_x(\tau)$ and in at most $n + 2\tau$ calls to the decision function δ_x (cf. Definition 2).*

The proof of Theorem 2 is provided in the appendix. In light of Theorem 1, we have the natural extension of Theorem 2 to the case of any leaking distance on $\mathbb{Z}_2^n$.

Corollary 1. *For any leaking distance d on $\mathbb{Z}_2^n$, Theorem 2 holds, with $\tau = h^{-1}(\tilde{\tau})$ being the corresponding threshold when $\tilde{\tau}$ is the given radius of the ball for the distance d.*

As a side result, we have:

Corollary 2. *If x is the stored biometric template b, and y is a matching fresh measurement b' satisfying $d(b, b') \leqslant \tau$, then Theorem 2 provides an algorithm to retrieve b being given b' in a number of authentication attempts linear in bit-length of the biometric templates.*

In the protocol for iris recognition by Daugman [5], the matching process relies on a normalised Hamming distance, which is defined as $\mathsf{NHD}(b, b', X, Y) = \sum_{i=1}^{n}(b_i \oplus b'_i)X_iY_i \,/\, \sum_{i=1}^{n} X_iY_i$, for $b, b', X, Y \in \mathbb{Z}_2^n$. In the previous formula the vector X is the mask for the stored biometric template b, while Y masks the fresh trait b'. It is immediate to see that the normalised Hamming distance does not comply with Definition 1, nevertheless it is still possible, given b' and Y, to mount a centre search attack and recover the bits of b that are not blinded by the mask X,*i.e.* b_i such that $X_i = 1$.

Theorem 2 holds only for leaking distances on $\mathbb{Z}_2^n$ as in the proof we exploit the fact that $|x_i - y_i|$ can only assume two values 0 and 1, when

$x_i = y_i$ and $x_i \neq y_i$ respectively. However, Theorem 3 generalises the reasoning in Theorem 2 to the non-binary case when any leaking distance is used (such as the Euclidean distance, often used in non-binary biometric authentication protocols).

Theorem 3. *Let $d : \mathbb{Z}_q^n \times \mathbb{Z}_q^n \to \mathbb{R}_{\geqslant 0}$ be any leaking distance on $\mathbb{Z}_q^n$ (cf. Definition 1) and $\tau \in \mathbb{R}_{>0}$, be a threshold such that $\tau < h(\lfloor \frac{q}{2} \rfloor^k)$, then it is possible to determine the value of the vector $x \in \mathbb{Z}_q^n$ having access only to a vector $y \in B_x(\tau)$ in at most mn calls to the decision function δ_x (as in Definition 2), where $m = \min\{\lfloor 2\tau \rfloor, 2 \log q\}$.*

The proof of Theorem 3 is provided in the appendix. Also in this case, if we consider the vectors as biometric templates it holds:

Corollary 3. *Considering x as the stored biometric template b, and y as the fresh matching trait b', then the proof of Theorem 3 provides an algorithm to mount centre search attacks against biometric authentication systems with templates in $\mathbb{Z}_q^n$. And the maximal number of authentication attempts is linear in length (dimension as vectors) of the biometric templates.*

It is important to highlight that the results of this section imply that all biometric authentication protocols that employ a leaking distance in the matching process are vulnerable to the *centre search* attack, and this attack can be performed in an efficient way.

5 Biometric Sample Recovery Attacks in the Binary Case

One of the most severe threats to biometric authentication systems is recovering a stored raw biometric template b (maybe linked to the identity of the user). The knowledge of b provides more information than the knowledge of a fresh trait b', as the same b could be used in multiple biometric authentication systems possibly employing different matching processes (while b' might be rejected). In Section 4 we already presented efficient ways to recover the centre b of a ball, given a point b' *close* to it, namely $b' \in B_b(\tau)$. The question we address now is: *Is there a way to find a matching template b' given access only to δ_b?* The next subsections present four different answers to this question. We discuss the connection between this problem and the *set-covering* problem in Section 5.2.

In the following, we consider only the case in which the biometric traits are binary vectors, *i.e.* $b \in \mathbb{Z}_2^n$, and the employed distance is a leaking distance (cf. Definition 1).

5.1 Blind Brute Force

In the *blind brute force attack*, the attacker randomly chooses a point $b' \xleftarrow{R} \mathbb{Z}_2^n$, and checks the output of the function $\delta_b(b')$. If $\delta_b(b') = 1$, it means that $p \in B_b(\tau)$, so the attacker can easily recover b using this point b' (cf. Theorem 2). Otherwise (*i.e.*, if $\delta_b(b') = 0$), the attacker picks another point at random from $\mathbb{Z}_2^n$ as before. We call this attack *blind brute force* because in each attempt the adversary tries a random point until a point in $B_b(\tau)$ is found.

Let us compute the success probability of this attack after $t \in \mathbb{Z}_{>0}$ attempts. Suppose first that we pick $b' \in \mathbb{Z}_2^n$ uniformly at random. Then the probability of having b' accepted is $\omega := |B_b(\tau)|/|\mathbb{Z}_2^n| = \sum_{k=0}^{\tau} \binom{n}{k}/2^n$. In each attempt, if the trial point is chosen uniformly at random and independently from the previous attempts, then with probability ω this new trial point will be accepted. Let us now introduce binary random variables $X_i = 0$ or 1, for $i = 1, 2, \cdots, t$, and let $\mathbb{P}(X_i = 1) = \omega$ and $\mathbb{P}(X_i = 0) = 1 - \omega$. Obviously, X_i, $i = 1, 2, \cdots, t$, are i.i.d. Bernoulli random variables $X_i \sim \mathcal{Bern}(\omega)$. We are interested in computing $\mathbb{P}\left(\sum_{i=1}^{t} X_i = 1\right)$, the total probability of succeeding once in t attempts. It is not hard to see that $\mathbb{P}\left(\sum_{i=1}^{t} X_i = 1\right) = t\omega(1-\omega)^{t-1}$, as the random variable $\sum_{i=1}^{t} X_i \sim \mathcal{Binom}(t, \omega)$ has a binomial distribution.

5.2 Sampling without replacement

Brute Force without Point Replacement In order to perform a brute force attack *without point replacement* the attacker has to define a set of potential candidates $C \subseteq \mathbb{Z}_2^n$. For the first trial, $C = \mathbb{Z}_2^n$ and the attacker chooses a point $b' \xleftarrow{R} C$ at random. If $\delta_b(b') = 1$, the selected point is inside the acceptance ball, $b' \in B_b(\tau)$, and so the attack is successful. Otherwise, the attacker updates the set of potential candidates $C = C \backslash \{b'\}$, deleting the one point that is not in the acceptance ball. The attack proceeds by randomly picking a point from the updated set C.

Let the random variables X_i, $i = 1, 2, \cdots, t$, be as in the case of the blind brute force attack. Note, however, that now $\mathbb{P}(X_i = 1)$ is different in each attempt. In this case, $\sum_{i=1}^{t} X_i$ follows the Hypergeometric distribution. Therefore, $\mathbb{P}\left(\sum_{i=1}^{t} X_i = 1\right) = B\binom{2^n - B}{t-1}/\binom{2^n}{t}$, where $B = |B_x(\tau)| = \sum_{k=0}^{\tau} \binom{n}{k}$. This attack is intuitively *better* than the blind brute force, but of course the larger the n is, the less efficient it is.

The Tree Algorithm We propose here a method (Algorithm 1) to find a point $b' \in \mathbb{Z}_2^n$ within distance τ from the unknown biometric template b, given access to the decision function δ_b (as in Definition 2). The central idea of Algorithm 1 is to consider the points of $\mathbb{Z}_2^n$ as leaves of a binary tree of depth n. The tree structure is then exploited to define relatives-relations among the points of $\mathbb{Z}_2^n$ and to ensure that at each unsuccessful trial one can delete non-overlapping portions of the space $\mathbb{Z}_2^n$. More precisely, if a point $p \in \mathbb{Z}_2^n$ is such that $\delta_b(p) = 0$, the algorithm removes from the set of potential centres not only the tried point p, but also its siblings-relatives generated by the τ common ancestor (see Figure 4).

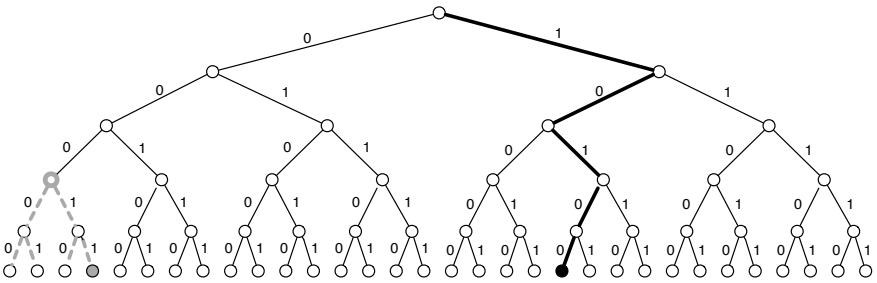

Fig. 2: The fundamental step of the Tree algorithm. Suppose the target biometric template is the vector $b = (10100) \in \mathbb{Z}_2^5$, the black bullet in the tree, and suppose the threshold is set to be $\tau = 2$. Let $a = (000)$ be the selected ancestor, highlighted as a grey circle in the picture. Let $b' = (00011)$ be the leaf randomly generated from a, then $d_H(b', b) > \tau$ and so $\delta_b\big((00011)\big) = 0$. In this case the points generated by a (*i.e.* that have a as common ancestor) will be deleted from the set of potential solutions.

The main function called by the algorithm is generate. Its input is the threshold τ and a $(n - \tau)$-dimensional binary vector a. The output is a random leaf $b' \in \mathbb{Z}_2^n$ generated by a (the τ ancestor). That is, $\mathsf{generate}(a, \tau) = (a_1, \ldots, a_{n-\tau}, r_1, \ldots, r_\tau) = b'$, where $r_i \in \mathbb{Z}_2, i = 1, \ldots, \tau$ are τ random bits. The set of potential ancestors C is updated at every unsuccessful round, by deleting the chosen ancestor. The threshold τ used in the tree algorithm is the Hamming distance threshold. This choice does not decrease the generality of the attack, as by Theorem 1 any (threshold $\tilde{\tau}$ for a) leaking distance can be written in terms of the Hamming one.

Algorithm 1 The Tree algorithm

Input: $(n, \tau, \delta_b,)$
Output: $b' = b'_1, \cdots, b'_n$ (a matching template)
$C = \mathbb{Z}_2^{n-\tau}$
for $i = 1$ to $2^{n-\tau}$: **do**
  $a \xleftarrow{R} \{C\}$
  $p = \mathsf{generate}(a, \tau)$
  if $\delta_b(b') = 1$ (accepted) **then**
    Return b'
  else
    $C = C \setminus \{a\}$
  end if
end for

For a practical implementation, one could store the paths of the tree that lead to the already rejected ancestors, and pick the new node a among the non-already-traversed paths. The running time of the attack is (of course) exponential, as it progressively constructs a binary tree of order $n - \tau$. Nevertheless, the probability to display the whole tree before finding a point that matches the reference template is very low (precisely: $2^{-n+\tau}$).

The *optimal* solution Recall that the goal of the attacks described in this section is to find the ball $B_b(\tau) \subset \mathbb{Z}_2^n$ on which δ_b takes the value 1, without any additional information at hand. We have already investigated blind brute force (random tries), brute force without point replacement (remove one point at each unsuccessful trial), and the Tree algorithm (remove 2^τ points at each unsuccessful trial). The *optimal* brute force approach exploits the following idea: if a point $p \in \mathbb{Z}_2^n$ is rejected, *i.e.* $\delta_b(p) = 0$, it means that $b \notin B_p(\tau)$. Hence, the whole ball $B_p(\tau)$ can be removed by the set of potential centres. Intuitively, the *best* one can do to rapidly reduce the size of potential centres, is to use as trial points, points that lie at distance 2τ from each other. This corresponds to covering the space $\mathbb{Z}_2^n$ with the *smallest number* of balls of radius τ. This corresponds to an instance of the well-known *set-covering* problem in a space [25, 26].

More precisely, the optimal biometric sample recovery attack would involve the adversary covering $\mathbb{Z}_2^n$ with a family $\mathfrak{F}$ of balls of radius τ. At this point, the adversary needs to query the oracle (*i.e.* to use the decision function δ_b) at most $|\mathfrak{F}|$ times, one for each (centre of a) ball in $\mathfrak{F}$. Hence the *best* solution is for $\mathfrak{F}$ a *minimal* covering, *i.e.* $|\mathfrak{F}| = min_{\mathfrak{G} \in \mathscr{C}} |\mathfrak{G}|$, where $\mathscr{C}$ is

the set of all possible covering of $\mathbb{Z}_2^n$ with balls of radius τ. This is exactly the set covering problem: to find the minimal number of balls needed to cover a space. It is proven that the set covering problem is NP-complete [26]. This result implies that also providing an *optimal* algorithm for the biometric sample recovery attack is an NP-complete problem. However, there exist some *greedy approximations* that are relatively efficient. In particular, for our case, Theorem 1 in [26] applies directly and hence the number of points that the adversary needs to query is only a factor of $O(\tau \ln(n+1))$ more than the optimal cover.

5.3 Comparisons and Bounds

In order to compare the performance of the four described methods we need to bound the probability that an attacker succeeds in finding a *matching* point, in each case. At the t-th trial, the attacker attempts point $x_t \in Z_2^n$ and observes $y_t \in 0, 1$, with $y_t \triangleq \mathbf{1}_{B_b(\tau)}(x_t) = \delta_b(x_t)$. Let $z_t \in \{0, 1\}$ denote whether or not the attacker has found an acceptable point after t trials and $s_t = \sum_{i=1}^{t} y_t$ be the number of points the attacker has found by time t.

To begin the analysis, we define $\mu_b(\tau) \triangleq |B_b(\tau)|/|\mathbb{Z}_2^n| \in [0, 1]$ to be the relative measure of the acceptance ball around b. In the binary case, dropping the dependence on b, τ, we have $\mu \in [2^{\tau-n}, (n+1)^\tau 2^{-n}]$. Of course, μ is also the probability of acceptance if sampling uniformly.

Blind brute force. In this case the points are selected uniformly without replacement, *i.e.* $x_t \sim \mathcal{U}(Z_2^n)$. It trivially follows that $\mathbb{E}(s_t) = \mu t$. It is also clear that the attack is successful whenever $s_t \geqslant 1$. For that reason, we shall attempt to bound the probability that this occurs while $\mu t < 1$. As a matter of fact, we can write:

$$\mathbb{P}(s_t \geqslant 1) = \mathbb{P}(\bigvee_{i=1}^{t} z_t = 1) \leqslant \sum_{i=1}^{t} \mathbb{P}(z_t = 1) = \mu t \leqslant (n+1)^\tau 2^{-n} t.$$

where the first inequality becomes an equality whenever $\mu t < 1$.

Sampling without replacement. All the other described approaches correspond to sampling without replacement. In either case, let $\alpha \in [0, 1]$ denote the proportion of points removed at each step. Then, we obtain the following bound:

$$\mathbb{P}(s_t \geqslant 1) \leqslant \sum_{i=1}^{t} \mathbb{P}(z_t = 1) \leqslant \sum_{i=1}^{t} \frac{\mu}{1 - \alpha i} \leqslant \int_0^t \frac{\mu}{1 - \alpha x}\, dx = \frac{\mu}{\alpha} \log \frac{1}{1 - \alpha t}.$$

For the point-wise replacement algorithm, $\alpha = q^{-n}$, hence there is little effect. For the binary case, we can employ the tree algorithm, $\alpha = 2^{\tau - n}$, which can be a substantial improvement. An unbounded adversary may use an optimal cover, in order to exclude as many points as possible whenever a point is rejected. In fact, in the best case, the adversary will be able to remove B points every time a point is rejected, giving a value of $\alpha = B2^{-n}$. To visualise the bounds, we choose some parameters such that there is a clear difference after a small number of iterations (depicted in Figure 3). More precisely, Figure 3 shows the performance of all four methods in terms of an upper bound on their success probability after a number of iterations. The four curves show sampling with replacement (*i.e.* brute force), and three different cases for sampling without replacement. Firstly, removing a single point. Secondly, removing 2^{τ} points using the tree construction. Finally, removing the maximum number of points B, which is computationally infeasible. There is a significant gain for the last choice, but only after a large portion of the space has already been covered. As when $\alpha \to 0$, $\ln \frac{1}{1-\alpha t} \to \alpha t$, the success probabilities of the first three methods are approximately linear in the size of the space, and hence exponential in the dimension.

The naive no replacement algorithm naturally does not improve significantly over brute force without replacement, since the volume that is excluded at every step is infinitesimal. Obviously, if we are able to remove a significant part of the volume, then we obtain a clear improvement in performance. Only an optimal adversary can do significantly better. However, this would assume either that *set-covering* is in P or that the adversary is computationally unbounded. Consequently, as there is no polynomial algorithm that is significantly better than brute force, biometric authentication schemes based on matching templates are secure against biometric sample recovery attacks.

6 Conclusions

In this paper, we prove that all biometric authentication protocols that employ distances between a template and an fresh biometric in the matching process suffer from leakage of information that could be exploited by an adversary to launch *centre search* attacks. In order to analyse this leakage of information, we provide a mathematical framework and prove that centre search attacks are feasible for any biometric template defined in $\mathbb{Z}_q^n$, $q \geqslant 2$, after a number of authentication attempts that is linear in n. Our results imply that it is possible to mount this attack on most

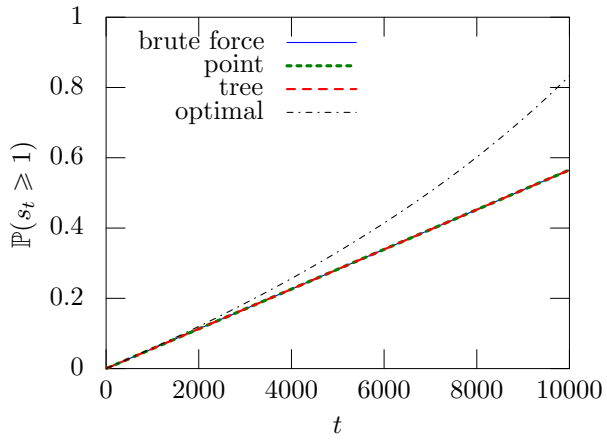

Fig. 3: Visualisation of the bounds for $q = 2$, $n = 32$, $\tau = 5$. In this case $\mu \approx 5.6 \times 10^{-5}$.

existing biometric authentication protocols (including privacy-preserving ones) that rely on a Hamming, Euclidean, normalised Hamming distance or any distance that complies with Definition 1.

Furthermore, we investigate whether brute force attacks can be used to recover a matching biometric. We describe four strategies: blind brute force, brute force without replacement, a new algorithm based on a tree structure and the optimal case. Our results demonstrate that improving the success rate in these brute force attacks would imply finding a solution to the NP-complete *set-covering* problem. Thus, this provides some security guarantees of existing biometric authentication protocols as long as the attacker has not access to a matching biometric trait.

A possible countermeasure that could be employed in order to strengthen existing biometric authentication protocols against *centre search* attacks would be the employment of more sophisticated authentication methods. For example, simply using weighted distances in which the weights are secret and different for each user may provide sufficient security. Something similar is already employed in the normalised Hamming distance for which indeed the centre search attack is feasible but only for a subset of the components of the stored biometric template. An alternative and promising direction would be to rely on a mechanism that randomly selects a distance from a pool of distances at each authentication attempt. However, such measures should be incorporated carefully in order not to affect the accuracy of the biometric authentication system.

Acknowledgements We would like to thank the anonymous reviewers for their comments. This work was supported by the FP7-STREP project "BEAT: Biometric Evaluation and Testing", grant number: 284989.

References

1. Penrose, L.: Dermatoglyphic topology. Nature **205** (1965) 544–546
2. Bolling, J.: A window to your health. Jacksonville Medicine, Special Issue: Retinal Diseases **51** (2000)
3. Osadchy, M., Pinkas, B., Jarrous, A., Moskovich, B.: SCiFI - A System for Secure Face Identification. In: Security and Privacy, 2010 IEEE Symposium on. (2010) 239–254
4. Bringer, J., Chabanne, H., Izabachène, M., Pointcheval, D., Tang, Q., Zimmer, S.: An application of the goldwasser-micali cryptosystem to biometric authentication. In: ACISP 2007. LNCS, Springer-Verlag (2007) 96–106
5. Daugman, J.: How iris recognition works. IEEE Transactions on Circuits and Systems for Video Technology **14** (2004) 21–30
6. Erkin, Z., Franz, M., Guajardo, J., Katzenbeisser, S., Lagendijk, I., Toft, T.: Privacy-preserving face recognition. In: Privacy Enhancing Technologies. (2009) 235–253
7. Sadeghi, A.R., Schneider, T., Wehrenberg, I.: Efficient privacy-preserving face recognition. In: ICISC 2009. LNCS (2009) 229–244
8. Huang, Y., Malka, L., Evans, D., Katz, J.: Efficient privacy-preserving biometric identification. In: NDSS 2011. (2011)
9. Barni, M., Bianchi, T., Catalano, D., Di Raimondo, M., Donida Labati, R., Failla, P., Fiore, D., Lazzeretti, R., Piuri, V., Scotti, F., Piva, A.: Privacy-preserving fingercode authentication. In: Proceedings of the 12th ACM workshop on Multimedia and security. (2010) 231–240
10. Simoens, K., Bringer, J., Chabanne, H., Seys, S.: A framework for analyzing template security and privacy in biometric authentication systems. IEEE Transactions on Information Forensics and Security **7**(2) (2012) 833–841
11. Jain, A.K., Nandakumar, K., Nagar, A.: Biometric template security. EURASIP J. Adv. Signal Process **2008** (2008) 113:1–113:17
12. Bringer, J., Chabanne, H., Simoens, K.: Blackbox security of biometrics. In: Proceedings of the 6th International Conference on Intelligent InformationHiding and Multimenida Signal Processing. (2010) 337–340
13. Jarrous, A., Pinkas, B.: Secure hamming distance based computation and its applications. In: ACNS 2009. Volume 5536 of LNCS. (2009) 107–124
14. Bringer, J., Chabanne, H., Patey, A.: SHADE: Secure hamming distance computation from oblivious transfer. In: Financial Cryptography Workshops. (2013) 164–176
15. Bringer, J., Chabanne, H., Favre, M., Patey, A., Schneider, T., Zohner, M.: GSHADE: Faster Privacy-preserving Distance Computation and Biometric Identification. In: Proceedings of the 2nd ACM Workshop on Information Hiding and Multimedia Security, ACM (2014) 187–198
16. Biham, E., Shamir, A.: Power analysis of the key scheduling of the aes candidates. In: Proceedings of the 2nd AES Candidate Conference. (1999)
17. Brier, E., Clavier, C., Olivier, F.: Correlation power analysis with a leakage model. In: CHES 2004. Volume 3156 of LNCS. Springer Berlin Heidelberg (2004) 16–29

18. Barbosa, M., Brouard, T., Cauchie, S., Sousa, S.M.: Secure biometric authentication with improved accuracy. In Mu, Y., Susilo, W., Seberry, J., eds.: ACISP 2008. Volume 5107 of LNCS., Springer (2008) 21–36
19. Stoianov, A.: Security issues of biometric encryption. In: Proceedings of the 2009 IEEE Toronto International Conference on Science and Technology for Humanity (TIC-STH). (2009) 34–39
20. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: EUROCRYPT 1999. Volume 1592 of LNCS. Springer (1999) 223–238
21. Yao, A.C.C.: How to generate and exchange secrets. In: Foundations of Computer Science, 1986., 27th Annual Symposium on, IEEE (1986) 162–167
22. Rabin, M.O.: How to exchange secrets with oblivious transfer. IACR Cryptology ePrint Archive **2005** (2005) 187
23. Goldwasser, S., Micali, S.: Probabilistic encryption and how to play mental poker keeping secret all partial information. In: Proceedings of the 14th Annual ACM Symposium on Theory of Computing. STOC 1982, ACM (1982) 365–377
24. Bringer, J., Chabanne, H., Cohen, G., Kindarji, B., Zémor, G.: Optimal iris fuzzy sketches. In: Proceedings of the 1st IEEE International Conference on Biometrics: Theory, Applications, and Systems. (2007)
25. Chen, L.: New analysis of the sphere covering problems and optimal polytope approximation of convex bodies. Journal of Approximation Theory **133**(1) (2005) 134–145
26. Chvatal, V.: A greedy heuristic for the set-covering problem. Mathematics of Operations Research **4**(3) (1979) pp. 233–235
27. Adler, A.: Vulnerabilities in biometric encryption systems. In: Audio-and Video-Based Biometric Person Authentication. Volume 3546 of LNCS, Springer 1100–1109

A Appendix

Below we provide the proof of Theorem 1:

Proof. By hypothesis d is a leaking distance, hence it is of the form $d(x,y) = h(\sum_{i=1}^{n} |x_i - y_i|^k)$, for all $x, y \in \mathbb{Z}_2^n$. Since $h : \mathbb{R} \to \mathbb{R}_{\geqslant 0}$ is monotonic, it is bijective on its image, in other words it has an inverse $h^{-1} : I \to \mathbb{R}$, where $I = Im(h) = \{w \in \mathbb{R}_{\geqslant 0} \ : \ w = h(z), \exists\, z \in \mathbb{R}\}$.
Consider the d ball of radius $\tilde{\tau}$ around a point $x \in \mathbb{Z}_2^n$, namely the set $\{y \in \mathbb{Z}_2^n \ : \ d(x,y) \leqslant \tilde{\tau}\}$. We want to prove this d-ball equals a *Hamming distance*-ball centred in x and of radius τ.
Indeed, $d(x,y) \leqslant \tilde{\tau} \iff h(\sum_{i=1}^{n} |x_i - y_i|^k) \leqslant \tilde{\tau}$. Noticing that h is increasing implies that h^{-1} is also increasing, one obtains: $\sum_{i=1}^{n} |x_i - y_i|^k \leqslant h^{-1}(\tilde{\tau})$. In addition, since $|x_i - y_i| \in \{0,1\}$ we can *ignore* the exponent k in the expression (this is because $0^k = 0$ and $1^k = 1, \forall\ k \in \mathbb{Q}_{>0}$). Hence, $\sum_{i=1}^{n} |x_i - y_i| \leqslant h^{-1}(\tilde{\tau})$, but the left hand side of the inequality is exactly the Hamming distance between the points x and y.
To summarise, we have $d(x,y) \leqslant \tilde{\tau} \iff d_H(x,y) \leqslant h^{-1}(\tilde{\tau})$. Let us put $\tau = h^{-1}(\tilde{\tau})$, then $\{y \in \mathbb{Z}_2^n \ : \ d(x,y) \leqslant \tilde{\tau}\} = \{y \in \mathbb{Z}_2^n \ : \ d_H(x,y) \leqslant \tau\}$.

That is, any d-ball can be described as a d_H-ball (*Hamming distance*-ball) and vice versa. □

Before we proceed with the proof of Theorem 2 and 3, let us briefly explain how the *hill climbing* attack works. In this attack, the adversary, in each trial, makes a small change to a forged biometric sample and observes how the matcher responds, which in our case is 1 or 0 corresponding to YES or NO. In doing so, the adversary can recover a biometric template that matches a stored template, after a number of trials; see [27] for details. In our case, the adversary has a matching template and wants to recover the stored template. So the attacker can use a variant of the hill climbing attack whereby he makes incremental changes to the matching template so that it moves to the boundary of the ball with center as the matching template and radius τ, the authentication threshold. Then by observing the matcher's response to the changes he makes to the template on the boundary, the attacker can recover the stored template component-by-component.

The proof of Theorem 2 is presented below:

Proof.
STEP 1. Find a point w that lies just outside the boundary of $B_x(\tau)$.
By hypothesis $\delta_x(y) = 1$. Let w be the vector obtained from y by flipping the first bit, *i.e.* $w_1 = \bar{y}_1$ and $w_i = y_i\,,\ \forall i \in \{2, \ldots, n\}$. If w is rejected, that is, if $\delta_x(w) = 0$, it means that y is already on the boundary of $B_x(\tau)$ and we are done by putting $v = y$. Otherwise, proceed by flipping one more bit of y until it exits $B_x(\tau)$. The general step after $k-1$ trials (flipping bits of y and being accepted) is: set $w = (\bar{y}_1, \ldots, \bar{y}_k, y_{k+1}, \ldots, y_n)$, if $\delta_x(w) = 0$ put $v = (\bar{y}_1, \ldots, \bar{y}_{k-1}, y_k, \ldots, y_n)$. If $\delta_x(w) = 1$, go on and flip the next component. It is quite intuitive that this procedure ends after at most $2\tau + 1$ steps (the worst case is when y is already on the boundary but we move it in the *wrong* direction and cross the ball along its diameter).

STEP 2. Determine the central point x of $B_x(\tau)$.
Note that by STEP 1, we already know the value of the k-th component of x, namely $x_k = v_k$. For $j \in \{1, 2, \ldots, n\}\backslash\{k\}$, consider the vector $v(j)$ defined as $v(j)_i = w_i,\ \forall\, i \in \{1, \ldots, n\}\backslash\{j\}$. If $\delta_x(v(j)) = 1$, it means that $v(j)$ *compensates* the error (in the k-th component) introduced by w with a *new* correct component (the j-th component). Hence $x_j = v(j)_j$. On the other hand, $\delta_x(v(j)) = 0$ implies that the j-th component of w was correct. Hence, in this case, $x_j = 1 - v(j)_j$. STEP 2 ends after $n-1$ queries. □

The proof of Theorem 3 is as follows:

Proof. Let $e(i) \in \mathbb{Z}_q^n$ denote the i-th vector of the canonical basis, *i.e.* for each $i = 1, \ldots, n$, $e(i)_i = 1$ and $e(i)_j = 0$, $\forall j \in \{1, \ldots, n\} \backslash \{i\}$. For each of the n components of a biometric template, determine two vectors $v(i), w(i) \in \mathbb{Z}_q^n, i = 1, \ldots, n$ such that: $v(i) = b' + \lambda_1 e(i)$ and $w(i) = b' + \lambda_2 e(i)$, with $\lambda_1 \in \{y_i, q-1-y_i\}$ and $\lambda_2 \in \{0, y_i - 1\}$. Moreover, $\delta_x\big(v(i)\big) = 1$ but $\delta_x\big(v(i) + e(i)\big) = 0$, and $\delta_x\big(w(i)\big) = 1$ but $\delta_x\big(w(i) - e(i)\big) = 0$. Such pair of vectors exists for each component, as $B_x(\tau)$ is a bounded subset of $\mathbb{Z}_q^n$ and $\tau < h(\lfloor \frac{q}{2} \rfloor^k)$. There are two possible situations:

- $v(i)$ and $w(i)$ are on the boundary of the ball $B_x(\tau)$. In this case the centre of the ball $x \in \mathbb{Z}_q^n$ will have the i-th component equal to the *middle point* $x_i = (v(i)_i + w(i)_i)/2$, $\forall i \in \{1, \ldots, n\}$.
- $v(i)$ and $w(i)$ are not exactly *on* the boundary of the ball $B_b(\tau)$. Since it is $v(i), w(i), x \in \mathbb{Z}_q^n$ the respective distances from the boundary $\epsilon_{v(i)}$ and $\epsilon_{w(i)}$ must be equal (by symmetry). Thus, also in this case $b_i = (v(i)_i + w(i)_i)/2$, $\forall i \in \{1, \ldots, n\}$.

There are two efficient strategies to determine the vectors $v(i)$, $w(i)$:

- *Linear search:* in this case the worst case scenario is when $y = x$, and the adversary needs to try all the points (with components in $\mathbb{Z}_q$) that lie in the diameter of the ball $B_x(\tau)$, that is at most $\lfloor 2\tau \rfloor$ trials.
- *Binary search:* the adversary performs at most $2 \log q$ trials to determine each *external* point, $v(i), w(i)$.

Thus, the maximum number of queries (access to the δ_x function) necessary in order to recover the centre x of a ball in $\mathbb{Z}_q^n$ is bounded by nm, with $m = \min\{\lfloor 2\tau \rfloor, 2 \log q\}$. □

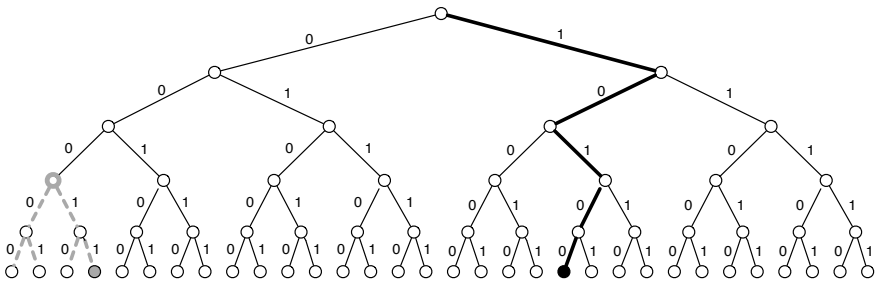

Fig. 4: The fundamental step of the Tree algorithm. Suppose the target biometric template is the vector $b = (10100) \in \mathbb{Z}_2^5$, the black bullet in the tree, and suppose the threshold is set to be $\tau = 2$. Let $a = (000)$ be the selected ancestor, highlighted as a grey circle in the picture. Let $b' = (00011)$ be the leaf randomly generated from a, then $d_H(b', b) > \tau$ and so $\delta_b\big((00011)\big) = 0$. In this case the points generated by a (*i.e.* that have a as common ancestor) will be deleted from the set of potential solutions.