Theoretical Analysis of Configurable RO PUFs and Strategies to Enhance Security

Theoretical Analysis of Configurable RO PUFs and Strategies to Enhance Security

Jiang Li[1], Hao Gao[1], Yijun Cui[1], Chenghua Wang[1] Yale Wang[1], Chongyan Gu[2] and Weiqiang Liu[1]
[1]College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China
[2]Centre for Secure Information Technologies, Queens University Belfast, Belfast, UK
Email: liuweiqiang@nuaa.edu.cn

***Abstract*—Compared to traditional ring oscillator PUF (RO PUF), configurable RO PUF (CRO PUF) greatly increases the number of challenge response pairs (CRPs) and improves hardware utilization. However, in the conventional CRO PUF design, when a path is selected by the challenge to generate a response, the circuit characteristic information constituting the CRO PUF, such as the delay information of the configurable unit, the transmission model, and etc., can also be leaked. Once the adversary monitors and masters this information, they can use this information to attack the CRO PUF circuits, such as modeling attacks. This paper establishes a theoretical model of CRO PUF and analyzes its unpredictability and security. Based on this model, a new mechanism to generate the proper challenges is proposed in this paper. In the proposed mechanism, the challenge is generated and utilized by a specific way, which can delay the feature leakage of the CRO PUF, thereby improving the security of the CRO PUF.**

***Index Terms*—Configurable RO PUF; modeling attacks; theoretical Analysis; secure improving strategies**

I. Introduction

THE physical unclonable function (PUF) is an important hardware security primitive that extracts uncontrollable manufacturing differences and generates a unique identity fingerprint for the circuit [1] [2]. Among various implementation principles, the ring oscillator-based physical unclonable function (RO PUF) [3] [4] is widely welcomed for its good performance and easy implementation. In order to improve the resource utilization of its application range, some researchers proposed some improved design CRO PUF of RO PUF based on the idea of reconfiguring different paths [5]. Although the number of CRPs increases and hardware consumption decreases, the correlation between them also increases significantly, which make them vulnerable of to potential attacks. When obtaining sufficient CRPs, machine modeling attacks can predict all CRPs and make the structure clonable [6] [7].

In order to improve the unpredictability of RO PUF and CRO PUF, the preliminary work mainly studied the key factors affecting the performance of ring oscillator based PUF [8], and proposed variety of frequency sorting algorithms [9]. Maiti and Schaumont proposed a method for selecting adjacent RO units for frequency comparison to compensate for the adverse effects of predictable process differences on the uniqueness of RO PUF [10]. The frequency distribution characteristics of RO arrays are analyzed, and two new oscillation frequency comparison strategies are proposed in [11]. However, in the CRO PUF designs, the challenge signals are consisted by two parts, one part is the configuring signal to organize the structure inside the CRO array and the other is the selecting signal to choose the CRO array [12]. Previous researches mainly focused on the later one and ignore the configuring inside the CRO array.

In this paper, the security of CRO PUF is analyzed by mathematical model [13], and the influence of configuring delay path is studied. A typical CRO PUF design representing these two main configuration methods was modeled and analyzed. An improved mechanism is proposed through the model.

II. Probability Models for CRO PUF

The ring oscillator is consisted by odd number of inverters [14]. There are subtle differences between these inverters due to uncontrolled manufacturing process variations. The delay difference of the RO causes the frequency difference. According to [15], the delay of the inverter follows a Gaussian distribution. In the CRO PUF design, the delay unit consists of multiple delay elements, either an inverter or a multiplexer. The delay profile of these elements can also be modeled using Gaussian distribution.

$$d_e \sim N(\mu_e, \sigma^2) \tag{1}$$

where d_e is the nominal value of the element delay time. The σ is the variance of the element delay time. Then the delay distribution n units is d_n, which can be express as:

$$d_n \sim N(n\mu_e, n\sigma^2) \tag{2}$$

Although the delay unit has different mean values due to different positions in the hardware, many literatures such as Liu and Maiti proposed appropriate RO comparison strategies [11] [16], which can make this difference ignored.Then the delay difference distribution D between the two delay units is:

$$D \sim N(0, n\sigma^2) \tag{3}$$

$Q(\cap_{k=1}^{n} CRP_k)$ indicates that the challenge k ($k = 1, \ldots n$) has been used to obtain their response (Q represents these CRPs which are known), and the attacker knows these CRPs. If the PUF server stores n CRPs, then the response i (CRP_i) generated by i is excited, and the unpredictability in the case of all CRPs except i is:

$$U_{pre_i} = P(D_i > 0 | Q(\cap_{k=1}^{n,k\neq i} CRP_k)) \tag{4}$$

The ideal value for CRP_i unpredictability is 0.5. The closer to 0.5, the better the unpredictability of CRP_i. Unpredictability of PUF n CRPs can be expressed as:

$$U_{pre_{PUF}} = max|P(D_i > 0|Q(\cap_{k=1}^{n,k\neq i} CRP_k)) - 0.5| \quad (i = 1, \cdots, n) \tag{5}$$

It is not only related to the design of the PUF, but also to the CRPs stored in the server. When CRPs are independent of each other, attackers cannot use the CRPs they have obtained to predict the unknown CRPs.

Assume that the attacker can obtain a small portion of the CRPs and unpredictability of CRP_i is also affected by other CRPs. In this paper, to simplify the analysis, the unpredictability of CRP_i based on the leaked CRP_k is described as:

$$U_{pre_i} = P(D_i > 0|Q(CRP_k))(k \neq i) \tag{6}$$

III. Security Analysis of the CRO PUF

According to the configuration strategy and CRO PUF structure [17], previous configurable designs can be classified as non-fixed frequency CRO PUF and fiexed frequency CRO PUF.

1) Non-fixed frequency CRO PUF: The non-fixed frequency CRO PUF is based on the number of delay elements. By selecting the input or non-input of the delay element to change the total number of stages to achieve the purpose of reconstruction, the frequency has a certain degree of change.

2) Fixed frequency CRO PUF: The fixed frequency CRO PUF is based on different delay elements and transmission path. By selecting different components on different paths for reconstruction, the total number of delay elements is constant. So compared to non-fixed frequency PUF, its frequency changes are small.

A. Security Analysis for Non-Fixed Frequency CRO PUF

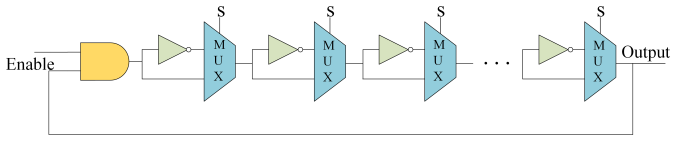

Fig. 1: Non-fixed frequency CRO PUF model.

A typical non-fixed frequency CRO PUF design is shown in Fig. 1 [14] [18]. In Fig. 1, the delay unit consists of a series of delay elements, each of which consists of an inverter and a MUX. Part of the challenge is designed to choose whether the inverter is chosen to construct the ring oscillator. Another part of the challenge is utilized to choose which two pairs of ROs to compare the frequencies. In order to ensure that the pair of delay units have symmetrical circuits, the same configuration of the paths of the two delay units is required. This reconstruction method is implemented by changing the number of delay elements in their delay units.

As mentioned above, the delay difference of this pair is subject to Gaussian distribution. When the cells with n delay elements are configured for frequency comparison, the result is $D_n > 0$. Then we assume that the attacker already knows the CRP_n. Next, the same pair is configured as $r = n-m$ delay elements, where m elements of n are selected not to input, and the frequency comparison result probability is no longer 0.5. Let D_r be the delay difference of the RO pairs of r elements, and D_n be the delay difference of the RO pairs composed of n elements.

Then the unpredictability of the CRP_r can be described as:

$$U_{pre_r} = P(D_r > 0|Q(CRP_n)) \tag{7}$$

$$P(D_r > 0|D_n > 0) = \frac{P(D_r > 0, D_r + D_m > 0)}{P(D_n > 0)} = 2\int_0^{+\infty} \frac{1}{\sqrt{2\pi r}\sigma} exp(-\frac{x^2}{2r\sigma^2}) \int_{-x}^{+\infty} \frac{1}{\sqrt{2\pi m}\sigma} exp(-\frac{y^2}{2m\sigma^2})\, dy\, dx \tag{8}$$

If the delay comparison result of configuring n delay elements is known, assuming that the result is $D_n < 0$, the distribution of D_r under this condition is:

$$\begin{aligned} f_r(x) &= \frac{2}{\sqrt{2\pi r}\sigma} exp(-\frac{x^2}{2r\sigma^2}) \int_{-x}^{+\infty} \frac{1}{\sqrt{2\pi m}\sigma} exp(-\frac{y^2}{2m\sigma^2})\, dy \\ &= \frac{1}{\sqrt{2\pi r}\sigma} exp(-\frac{x^2}{2r\sigma^2}) erfc(-\frac{x}{\sqrt{2\pi m}\sigma}) \end{aligned} \tag{9}$$

Similarly, the distribution of D_r under $D_n > 0$ is:

$$\begin{aligned} f_r(x) &= \frac{2}{\sqrt{2\pi r}\sigma} exp(-\frac{x^2}{2r\sigma^2}) \int_{-\infty}^{-x} \frac{1}{\sqrt{2\pi m}\sigma} exp(-\frac{y^2}{2m\sigma^2})\, dy \\ &= \frac{2}{\sqrt{2\pi r}\sigma} exp(-\frac{x^2}{2r\sigma^2})(1 - \frac{1}{2} erfc(\frac{-x}{\sqrt{2\pi m}\sigma})) \end{aligned} \tag{10}$$

Fig. 2 and Fig. 3 show the variation of the D_r distribution. If $D_n > 0$ or $D_r < 0$, the distribution of D_r is shifted from the Gaussian distribution to the $+x$ or $-x$ direction. In the two distribution curves, the variance σ and the initial number n of delay elements are constant. When m changes from 1 to 5, the slope also becomes larger.

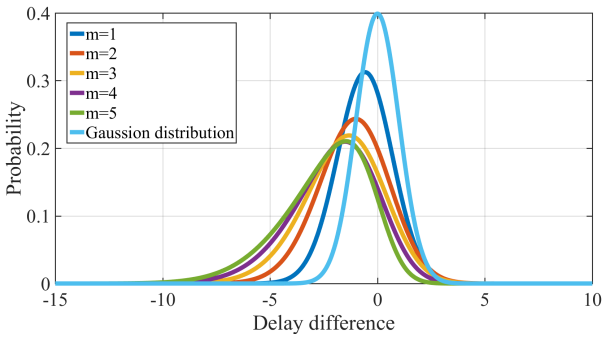

Fig. 2: The variation of the D_r distribution when $D_n > 0$ is known (m is the elements that are selected not to input).

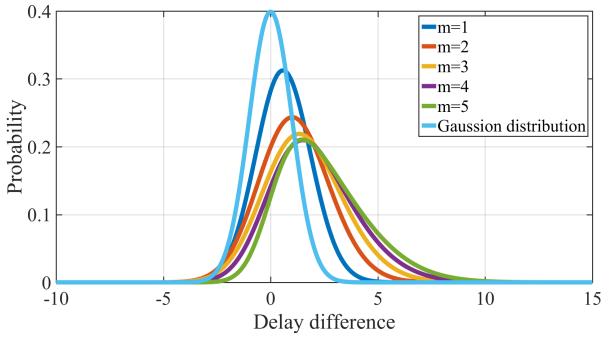

Fig. 3: The variation of the D_r distribution when $D_n < 0$ is known (m is the elements that are selected not to input).

Since the probability distributions of D_r are known under known D_n conditions, we can calculate the probability of D_r. Suppose the current number of delay elements is r, the original number is n, and the number of reductions or increases is m. Fig. 4 shows the effect of the ratio of r and n ($\frac{r}{n}$) on its probability. Under the condition $D_n > 0$, when $\frac{r}{n}$ tends to 1, the probability of $D_r < 0$ tends to 0, indicating that CRP_r has a strong correlation with CRP_n; when $\frac{r}{n}$ approaches 0, the probability of $D_r < 0$ tends to 0.5, indicating that the correlation between CRP_r and CRP_n is weak.

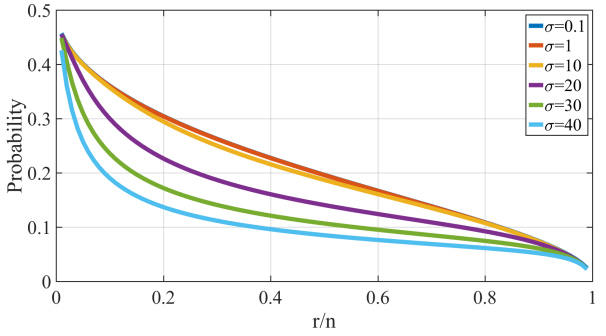

Fig. 4: The effect of $\frac{r}{n}$ on $D_r < 0$ probability ($\frac{r}{n}$ is the ratio of the current number and original number of delay elements and the σ is the variance of the element delay time).

In order to improve the unpredictability of non-fixed frequency PUF, it is necessary to remove CRPs that have a greater correlation with other CRPs in the database. This means that a pair of delay units can only use several configuration methods because of the strong correlation between different configuration methods in the same pair of delay units. In addition, reducing the delay variance of each element helps to improve the unpredictability of the PUF.

B. Security Analysis for Fixed Frequency CRO PUF

Cui proposed a fixed frequency CRO PUF based on the tristate inverters, which configures the delay unit by changing the delay elements in the delay unit [19] [20]. The tri-state matrix unit consists of an $n \times m$ CMOS tri-state gate array and

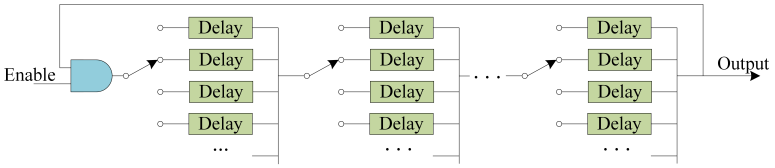

Fig. 5: Fixed frequency CRO PUF model.

a two-input NAND gate. This is a very representative fixed frequency PUF, and Fig. 5 shows its simplified model. When the enable is enabled, the RO oscillates with an oscillation frequency associated with the delay elements through which the signal passes. The challenge can select the delay elements through which the signal passes to achieve the purpose of configuring this RO.

If a delay unit contains 4×4 elements $C(i, j)$, where i and j vary from 1 to 4, and there is only one pair of delay units, $C(1,1), C(1,2), C(1,3)$ and $C(1,4)$ are configured as D_1. $C(2,1), C(2,2), C(2,3)$ and $C(2,4)$ are configured as D_2. $C(1,1), C(1,2), C(2,3)$ and $C(2,4)$ are configured as D_3. The first two elements of D_1 are denoted by $D_1^{'}$, the last two elements of D2 are by $D_2^{'}$, and then $D_3 = D_1^{'} + D_2^{'}$.

Obviously D_2 is independent of D_1. An attacker cannot predict D_2 from the results of D_1, and vice versa. But both D_1 and D_2 have the same path elements as D_3.They are related. Under the condition that $D_1 > 0$ is known, the probability of $D_3 > 0$ is:

$$\begin{aligned} U_{pre_{D_3}} =& P(D_3 > 0 | D_1 > 0) \\ =& \frac{D_3 > 0, D_1 > 0}{P(D_1 > 0)} \\ =& 2 \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi * 2l\sigma}} exp(-\frac{x_m^2}{4l\sigma}) * \\ & \int_{-x_l}^{+\infty} \frac{1}{\sqrt{2\pi * 2r\sigma}} exp(-\frac{x_r^2}{4r\sigma}) * \\ & \int_{-x_l}^{+\infty} \frac{1}{\sqrt{2\pi * 2r\sigma}} exp(-\frac{x_r^{'2}}{4r\sigma}) \, dx_r^{'} \, dx_r \, dx_m \end{aligned} \tag{11}$$

where l and r represent the number of identical and different elements in D_3 and D_1, respectively, and n is the total number of elements. In the above assumption, both l and r are equal to two. Fig. 6 shows the probability of $D_3 > 0$ for different $\frac{r}{n}$ ($n = r + l$) and sigma, and $\frac{r}{n}$ can represent the degree of difference between the two ROs. From Fig. 6, we can draw a similar conclusion to the non-fixed frequency PUF. Increasing the number of different elements between two ROs can reduce the correlation, but also reduce the resource utilization. In addition, reduce the variance of each element can improve the unpredictability.

However, unlike the non-fixed frequency PUF, the initial delay profile of $D_1^{'}$ or $D_2^{'}$ is performed obeys the Gaussian distribution before the comparison. While after the comparison, it is known that D_1 or D_2 is larger or smaller than 0, and the distribution of $D_1^{'}$ or $D_2^{'}$ will shift in the positive or negative x direction. The magnitude of the offset depends on

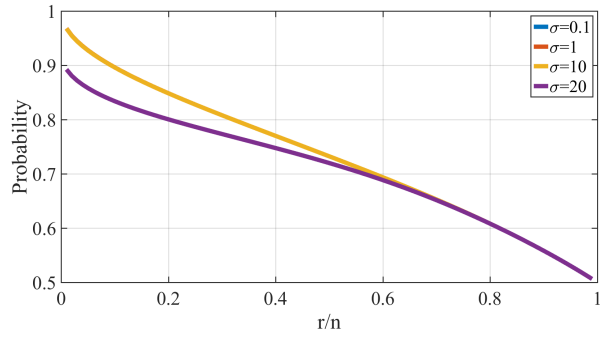

Fig. 6: The effect of $\frac{r}{n}$ on $D_3 > 0$ probability ($\frac{r}{n}$ is the ratio of the current number and original number of delay elements and the σ is the variance of the element delay time). The curves of σ=0.1, 1 and 10 are the same.

n, r, σ. The direction of the offset depends on the result of the comparison. If it is greater than 0, it is right-biased.

As shown in Fig. 7, under the condition that $D_1 > 0$ and $D_2 < 0$, $D_1^{'}$ and $D_2^{'}$ have the same distribution offset and opposite directions. The distribution of D_3 will cancel the offset between the right of $D_1^{'}$ and the left of $D_2^{'}$, so the delay difference is Gaussian distribution centered on 0.

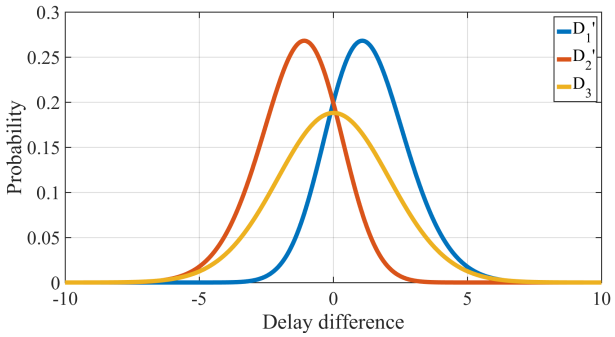

Fig. 7: The delay difference of $D_1^{'}$, $D_2^{'}$ and D_3.

As can be seen from the above, the delayed distribution of the compared elements obeys Gaussian distribution and has different offset factors. In D_3, if the delay distribution offset factors of the elements cancel each other, the comparison results will still maintain good unpredictability. Compared with non-fixed frequency PUF, this can improve unpredictability and improve hardware utilization.

In summary, when the two RO rings have too many identical delay elements, there is a great correlation between them. In order to increase the unpredictability, the challenge for this situation can be eliminated. However, in fixed frequency PUF, some challenge selection mechanisms can be used to allow certain CRPs to be used first, so that other CRPs will still be unpredictable. The fixed frequency PUF has more configuration paths and has good unpredictability.

IV. Configure Strategy to Enhance The Security

Because the challenges of the left and right deviations can be canceled at the same time. Therefore, when multiple challenges are selected, as long as the total left and right deviations of the challenges are the same, then the challenges will remain well unpredictable. Therefore combined with [21], this section proposed a algorithm to generate challenges to enhance security.

Assume that there are f RO pairs for frequency comparison, each RO pair having RO_a and RO_b, each RO having m-level delay, and each delay having n optional delay elements.

Definitions: (1) $n \times m$ matrix S_i represent the delayed state matrix of the i-th RO pair. The element in the p-th column and the q-th row in S_i corresponds to the state of the delay difference between delay elements of RO_a and RO_b on the q-th of the class p.

(2) $n \times m$ matrix C_{ij} represent the challenge matrix of the j-th challenge generated on the i-th RO pair. The element on the p-th column and q-th row in matrix C_{ij} corresponds to the q-th delay element on class p of RO_a and RO_b and the value of the element in C_{ij} is 0 or 1. 1 indicates that the corresponding delay unit is selected, while 0 indicates that it is not selected. There is one and only one 1 per column in C_{ij}.

(3) $n \times m$ matrix $\sim C_{ij}$ represent the matrix after the values of each element in C_{ij} are reversed by bits.

(4) Challenge matrix set CH_i, including all the challenges matrices of the i-th RO.

(5) The f-dimensional vector u, where $u(i)$ represents the number of challenges that have been used by the i-th RO pair.

(6) Operation matrix a $*$ matrix b = matrix c (C_{ij}=$a_{ij} * b_{ij}$).

(7) Operation $\bigsqcup C_{ij}$ is the comparison of the frequencies of RO_a and RO_b. If the result is 0, it represents that the frequency of RO_a is less than that of RO_b under the challenge of C_{ij}. On the contrary, the frequency of RO_a is greater than RO_b.

(8) Operation $\bigcup$ (matrix a) and the result of the operation is the value obtained by multiplying each non-zero element of the matrix a.

(9) Operation $\bigcap$ (matrix a) and the result of the operation is the value obtained by adding all the elements of the matrix a.

Algorithm 1 is the concrete algorithm. This algorithm is further illustrated by a concrete matrix as follows:

Initialization. Produces the $n \times m$ state matrix of the delay elements. The values of each element in the matrix correspond to the distribution state of the element delay difference in two identical ROs. A value of 1 indicates that the distribution of the corresponding element delay difference is Gaussian distribution with an average value of 0. A value of R^t represents the distribution of the Gaussian distribution with an average value of 0 after the right offset, and t is the degree offset of the process. A value of L^t represents the distribution of the Gaussian distribution with an average value of 0 after the left offset, and t is the degree of the offset. The initial value of

Algorithm 1: Challenge Generation Algorithm

Input: Matrix S_i $(0 < i \leq f)$, challenge matrix set CH_i, vector u, threshold k, repeat number r.
Output: challenges

initialization;
while *receive request for challenge* **do**
  Select the i-th RO pair randomly;
  $j = u(i) + 1$;
  Select a challenge matrix in CH_i randomly called C_{ij};
  if $CH_i = \varnothing$ **then**
    continue
  end
  if $\mid log_R \bigcup(C_{ij} * S_i) - 1 \mid < k$ **then**
    for $e = r; e \geq 0; e = e - 1$ **do**
      Select a challenge matrix in CH_i randomly called C_{ij};
      if $\mid log_R \bigcup(C_{ij} * S_i) - 1 \mid < k$ **then**
        break
      end
    end
    if$e < 0 \&\& \mid log_R \bigcup(C_{ij} * S_i) - 1 \mid > k$ continue
  end
  Output C_{ij} to get response and remove C_{ij} from CH_i;
  $u(i) = u(i) + 1$;
  if $\bigsqcup(Cij) == 0$ **then**
    $g = R$
  else
    $g = 1/R$
  end
  $S_i = (S_i * C_{ij}) \times g + (S_i * (\sim C_{ij}))$;
  $z = u(i) - 1$
  while $z > 0$ **do**
    $z = u(i) - 1$;
    $x = \bigcap(C_{ij} * C_{iz})$;
    $x = \bigcap(\sim C_{ij} * C_{iz})$;
    $S_i = (S_i * (C_{ij} * C_{iz})) \times \frac{x}{y} \times (\frac{1}{g})$;
    $z = z - 1$;
  end
end

the matrix D_1 is:

$$D_1 = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & 1 & 1 & \cdots & 1 \end{pmatrix}$$

Challenge Selection. Randomly selects one element from each column in the matrix to form an challenge. If the selected elements is multiplied, the result will be $R^t L^t$ (in the multiplication process, R and L cancel each other, that is, $R * L = 1$). When t is less than the set threshold, it indicates that the selected elements delay differential distribution offset to the right or left is not a lot and the challenge is feasible and can produce such challenge to use. Otherwise, it is not feasible and need to re-select the challenge of the Matrix.

Adjustment of Delay Difference State Matrix. The previous selected elements is multiplied by an offset factor based on the comparison result. If the first RO is greater than the second RO, it is the right deviation recorded as R. If the first RO is less than the second RO, it is left deviation recorded as L. The state matrix is then adjusted according to all the challenges previously generated. For the $n \times m$ matrix, if the challenge generation is recorded as:

For all elements of the first row, the comparison result is that the first one is greater than the second.

For all elements of the second row, and the comparison result is that the first one is less than the second.

The state matrix at this point is:

$$D_2 = \begin{pmatrix} R & R & R & \cdots & R \\ L & L & L & \cdots & L \\ 1 & 1 & 1 & \cdots & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & 1 & 1 & \cdots & 1 \end{pmatrix}$$

Then, if w elements from the front of the first row were selected, and the $m = n - w$ elements from the back of the second row will form an challenge. Multiply the elements into R^{w-m}, which can be used when the $2w - n$ size satisfies the condition of set threshold. After use, assuming that the that the result is that the first RO is greater than the second, we adjust the matrix, first multiply the elements inside the challenge by R.

$$D_3 = \begin{pmatrix} R^2 & R^2 & R^2 & \cdots & R \\ L & L & L & \cdots & LR \\ 1 & 1 & 1 & \cdots & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & 1 & 1 & \cdots & 1 \end{pmatrix}$$

According to the challenges used before this challenge, that is, the first line, the second line, we further adjust the matrix. These challenges that need to be adjusted for elements that have been used in this challenge, in addition to those elements used in this challenge and we multiply the other elements of those challenges by a reverse bias factor that is contrary to this bias. Following the above hypothesis, because this time is right deviation, the adjustment factor is the L. The index of L is the ratio of the selected elements and the unselected elements in the previous challenges. The first line is $\frac{w}{m}$ and the second line is $\frac{m}{w}$.

$$D_4 = \begin{pmatrix} R^2 & R^2 & R^2 & \cdots & RL^{\frac{w}{m}} \\ L^{1+\frac{m}{w}} & L^{1+\frac{m}{w}} & L^{1+\frac{m}{w}} & \cdots & LR \\ 1 & 1 & 1 & \cdots & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & 1 & 1 & \cdots & 1 \end{pmatrix}$$

Then repeat the Challenge Selection phase.

Use the Algorithm 1 to generate challenge. As shown in the Fig. 8, we can see that in the challenge state matrix, elements

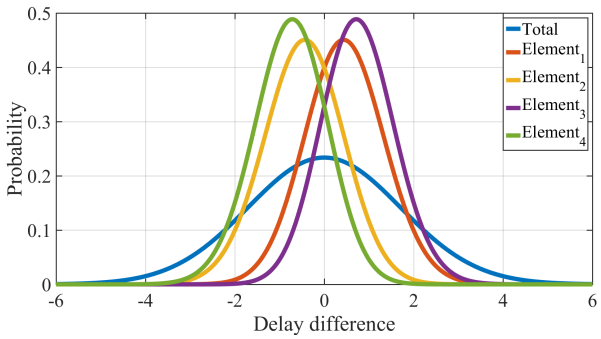

Fig. 8: Elimination of the offset (The left and right deviations are canceled at the same time).

with two rightwards bias (element 1 and element 3) and two leftwards bias (element 2 and element 4) are selected, and their overall degree of leftwards bias is the same as that of rightwards bias. The factors of left and right deviations in the response cancel each other out, then the delay difference of the final two ROs units remains unbiased and the unpredictability is still 0.5. Therefore, the use of this algorithm can indeed improve the unpredictability of CRPs, thereby improving the security of the PUF.

V. Conclusion

In this paper, a theoretical analysis for the CRO PUF is presented. We built the mathematic model for both non-fixed frequency CRO PUF and fixed frequency CRO PUF. The analysis and the deduction of the delay differences for the CRO PUF shows that the variance of each delay element has a negative impact on the unpredictability of the system. Moreover, the delay difference distribution of a specific pair of CRO PUF delay units will be influenced by the result of comparison of other known delay features (leaked from used CRPs), which will reduce the unpredictability of the CRO PUF. Based on the theoretical analysis, an improved mechanism to generate the proper challenges for the CRO PUF is proposed, which can delay the leakage of CRO PUF features and improve the unpredictability of non-fixed frequency CRO PUF. The analysis shows that the proposed mechanism can produce the challenge in a risk-controllable way and improve the security of the CRO PUF.

Acknowledgment

This work is supported by the State Grid Corporation Science and Technology Project Funded "Key Technology Research on Trustworthy Identity Authentication of Grid Core Business" (52110418001L), and the National Natural Science Foundation of China (61771239).

References

[1] R. Maes, *Physically unclonable functions: Constructions, properties and applications*. Springer Science & Business Media, 2013.

[2] H. Handschuh, G.-J. Schrijen, and P. Tuyls, "Hardware intrinsic security from physically unclonable functions," in *Towards Hardware-Intrinsic Security*, pp. 39–53, Springer, 2010.

[3] A. Maiti, J. Casarona, L. McHale, and P. Schaumont, "A large scale characterization of RO-PUF," in *Proc. IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pp. 94–99, 2010.

[4] J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls, "FPGA intrinsic PUFs and their use for IP protection," in *Proc. International Workshop on Cryptographic Hardware and Embedded Systems*, pp. 63–80, 2007.

[5] Y. Cui, C. Wang, W. Liu, Y. Yu, M. O'Neill, and F. Lombardi, "Low-cost configurable ring oscillator PUF with improved uniqueness," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 558–561, 2016.

[6] Q. Ma, C. Gu, N. Hanley, C. Wang, W. Liu, and M. O'Neill, "A machine learning attack resistant multi-puf design on FPGA," in *Asia and South Pacific Design Automation Conference*, 2018.

[7] Z. Lu, D. Li, H. Liu, M. Gong, and Z. Liu, "An anti-electromagnetic attack puf based on a configurable ring oscillator for wireless sensor networks," *Sensors*, vol. 17, no. 9, p. 2118, 2017.

[8] Y. Wang, C. Wang, C. Gu, Y. Cui, M. O'Neill, and W. Liu, "Theoretical analysis of delay-based pufs and design strategies for improvement," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, May 2019.

[9] Z. Zhang, C. Gu, Y. Cui, C. Zhang, M. O'Neill, and W. Liu, "Multi-Incentive Delay-Based PUF," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, May 2019.

[10] A. Maiti and P. Schaumont, "Improving the quality of a physical unclonable function using configurable ring oscillators," in *Proc. IEEE International Conference on Field Programmable Logic and Applications(FPL)*, pp. 703–707, 2009.

[11] W. Liu, Y. Yu, C. Wang, Y. Cui, and M. O'Neill, "RO PUF design in FPGAs with new comparison strategies," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 77–80, 2015.

[12] Z. Paral and S. Devadas, "Reliable and efficient PUF-based key generation using pattern matching," in *Proc. IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pp. 128–133, 2011.

[13] J. Delvaux, D. Gu, D. Schellekens, and I. Verbauwhede, "Helper data algorithms for PUF-based key generation: overview and analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 6, pp. 889–902, 2015.

[14] M. Gao, K. Lai, and G. Qu, "A highly flexible ring oscillator PUF," in *Proc. 51st ACM Annual Design Automation Conference*, pp. 1–6, 2014.

[15] R. Maes and I. Verbauwhede, "Physically unclonable functions: A study on the state of the art and future research directions," in *Towards Hardware-intrinsic Security, Security & Cryptology*, 2010.

[16] A. Maiti and P. Schaumont, "Improved ring oscillator PUF: an FPGA-friendly secure primitive," *Journal of cryptology*, vol. 24, no. 2, pp. 375–397, 2011.

[17] L. Zhang, C. Wang, W. Liu, M. O'Neill, and F. Lombardi, "XOR gate based low-cost configurable RO PUF," in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–4, 2017.

[18] Y. Lao and K. K. Parhi, "Statistical analysis of mux-based physical unclonable functions," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 33, no. 5, pp. 649–662, 2014.

[19] Y. Cui, C. Gu, C. Wang, M. ONeill, and W. Liu, "Ultra-lightweight and reconfigurable tristate inverter based physical unclonable function design," *IEEE Access*, vol. 6, pp. 28478–28487, 2018.

[20] Y. Cui, C. Wang, W. Liu, and M. O'Neill, "A reconfigurable memory PUF based on tristate inverter arrays," in *Proc. IEEE International Workshop on Signal Processing Systems (SiPS)*, pp. 171–176, 2016.

[21] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Proc. 44th ACM Annual Design Automation Conference*, pp. 9–14, 2007.