

Functional Specification for Visualizing PUF Data

Main Tasks

Scenario 1: Generation of the Response Matrix and Reordering.

Generation of the Response Matrix

- The user first inputs 2 values: 1) number of PUFs 2) number of challenges.
- The application will now generate δ values for each PUF. $2N$ δ values will be generated.
- The δ values will be randomly sampled from a **gaussian distribution** having mean = 0 and variance = 1.
- If the user wants to input their own δ values, we will provide an option to upload a JSON file contains δ values.
- After δ values are sampled and stored, the tool will now generate the response matrix.
- The response matrix will be generated using the recursive calculation as follows, where $\text{sign}(\Delta_n)$ is the final response.

$$\Delta_i(\mathbf{c}) = \begin{cases} +\Delta_{i-1}(\mathbf{c}) + \delta_i^{(0)}, & \text{when } c_i = 0 \\ -\Delta_{i-1}(\mathbf{c}) + \delta_i^{(1)}, & \text{when } c_i = 1 \end{cases} \quad (2)$$

Reordering according to bit position

- In order to reorder the rows of the response matrix, the user will input a value i (bit position)
- Then, the challenge set C will be partitioned into subsets $C_{i,+}^{(0)}, C_{i,-}^{(0)}, C_{i,+}^{(1)}, C_{i,-}^{(1)}$
- The rows of the response matrix will be reordered according to these subsets

Reordering according to δ values

- The user will enter 2 values, a bit position i and a bit value x .
- The columns of the response matrix will be reordered according to $\delta_i^{(x)}$ of each PUF.

Reference: [APUF Faults: Impact, Testing, and Diagnosis - Yeqi Wei, Tim Fox, Vincent Dumoulin, Wenjing Rao, Natasha Devroye](#)

Scenario 2 : Get Score of a selected area.

- The user will select a submatrix from the response matrix by using a brush interaction on the response matrix.
- Once the user selects the submatrix, the **score** of the submatrix will be displayed in an adjoining panel.
- The **score** is defined as the **response bias** $R(C)$ of that submatrix - **which is defined as the number of 1s in the submatrix.**
- Inside the selected area, the user has the ability to select any two APUFs i.e two row vectors. Then, we calculate the hamming distance of those 2 responses which is denoted by $HD_{ij}(C)$. This is also displayed in an adjoining panel.

Scenario 3: Generalize scenario 2 for a custom function

- The user wants to define their own function for calculating the **score** of a submatrix.
- The user then defines/selects a function $f(x)$ for calculating the **score** using some provided interface and proceeds as per scenario 2.

Scenario 4: Visualize DNA OF PUFs

- The user selects a PUF number from a dropdown and clicks on a button **View PUF Data**

- A visualization of the δ values of that PUF is displayed on screen.
- A **histogram** of the Δ values of that PUF is also displayed on screen.

Example plots:

Now I want PUF # 30
(row # 30 for the matrix)

