Question 1

Let's prove that a linear model can indeed predict the time taken for the cipher (or lower) signal to reach the finish line for a simple arbiter PUF.

In the question, we have 8 MUX for each PUF. Let's number them 0 to 7.

t_i^u, t_i^l = Time taken for the i^{th} MUX to generate the cipher & lower output signal.

$\Delta_i = t_i^u - t_i^l$ ⇒ Delay between the cipher and lower signals for the i^{th} MUX.

$t_{-1}^u = 0,\ t_{-1}^l = 0,\ \Delta_{-1} = 0$ ∴

From the slides, we know

$$t_i^u = (1-c_i)(t_{i-1}^u + p_i) + c_i(t_{i-1}^l + s_i) \quad \text{— (1)}$$

$$c_i = \frac{1+d_i}{2} \quad \text{— (2)}$$

From (1)

$$t_i^u = t_{i-1}^u + p_i - c_i t_{i-1}^u - c_i p_i + c_i t_{i-1}^l + c_i s_i$$

$$t_i^u = t_{i-1}^u + c_i\left[s_i - p_i + \Delta_{i-1}\right] + p_i \quad \text{— (3)}$$

Putting (2) in (3)

$$t_i^u = t_{i-1}^u + \left(\frac{1-d_i}{2}\right)\left[s_i - p_i + \Delta_{i-1}\right] + p_i$$

$$t_i^u = t_{i-1}^u + \left(\frac{p_i+s_i}{2}\right) + \frac{d_i}{2}\left[p_i - s_i\right] + \left(\frac{1+d_i}{2}\right)\Delta_{i-1}$$

Let $e_i = \frac{p_i+s_i}{2}$, $f_i = \frac{p_i-s_i}{2}$.

$$\therefore t_i^u = t_{i-1}^u + e_i + d_i f_i + \left(\frac{1-d_i}{2}\right)\Delta_{i-1}$$

For $i=0$.

$$t_0^u = c_0 + d_0 f_0 \quad \text{as } t_{-1}^u = 0 \text{ \& } \Delta_{-1} = 0$$

For $i=1$

$$t_1^u = c_0 + d_0 f_0 + c_1 + d_1 f_1 - \left(\frac{1-d_1}{2}\right)\Delta_0$$

$$t_1^u = (c_0 + c_1) + (d_0 f_0 + d_1 f_1) - \left(\frac{1-d_1}{2}\right)\Delta_0$$

For $i=2$

$$t_2^u = (c_0 + c_1 + c_2) + (d_0 f_0 + d_1 f_1 + d_2 f_2) - \frac{1}{2}\left[(1-d_1)\Delta_0 + (1-d_2)\Delta_1\right]$$

Continuing in the same fashion, we get.

$$t_7^u = \sum_{i=0}^{7}(c_i + d_i f_i) - \frac{1}{2}\left[\Delta_0 + \Delta_1 \cdots \Delta_6 - d_1\Delta_0 - d_2\Delta_1 \cdots - d_7\Delta_6\right]$$

$$t_7^u = \sum_{i=0}^{7}(c_i + d_i f_i) - \frac{1}{2}\left[(\Delta_6 - d_6\Delta_5) + (\Delta_5 - d_5\Delta_4) + \cdots (\Delta_1 - d_1\Delta_0) + \Delta_0 - d_7\Delta_6\right] \quad (4)$$

We know from slides,

$$\Delta_i = d_i\Delta_{i-1} + d_i d_i + \beta_i$$

$$\Delta_i - d_i\Delta_{i-1} = d_i d_i + \beta_i \quad \text{— (5)}$$

$$\Delta_0 = d_0 d_0 + \beta_0 \quad \text{— (6)} \quad \text{as } \Delta_1 = 0$$

$$d_7\Delta_6 = \Delta_7 - d_7 d_7 - \beta_7 \quad \text{— (7)}$$

Putting 5, 6, 7 in 4

$$t_7^u = \sum_{i=0}^{7}(c_i + d_i f_i) - \frac{1}{2}\left[\alpha_6 d_6 + \beta_6 + \alpha_5 d_5 + \beta_5 + \cdots \alpha_1 d_1 + \beta_1 + \alpha_0 d_0 + \beta_0 - \Delta_7 + \alpha_7 d_7 + \beta_7\right]$$

$$t_7^u = \sum_{i=0}^{7}(c_i + d_i f_i) - \frac{1}{2}\sum_{i=0}^{7}(d_i d_i + \beta_i) + \frac{\Delta_7}{2}$$

we know from slides

$\Delta_i = \sum_{j=0}^{i} w_j x_j + b$, $\Delta_d = \sum_{i=0}^{K-1} w_i x_i + b$

where x_j

where

$x_i = d_i d_{i+1} \cdots d_{K-1}$

$w_0 = \alpha_0$

$w_i = \alpha_i + \beta_{i-1}$ if $i > 0$

$b = \beta_{K-1}$

$\therefore t_7^u = \sum_{i=0}^{7} \left(e_i - \frac{\beta_i}{2}\right) + \sum_{i=0}^{7} \left(d_i f_i - \frac{\alpha_i d_i}{2}\right) + \frac{1}{2}\sum_{i=0}^{7} w_i x_i + \frac{\beta_7}{2}$

Inserting the expressions for e_i & f_i, we get.

$t_7^u = \sum_{i=0}^{7} \left(\frac{p_i + s_i - \beta_i}{2}\right) + \sum_{i=0}^{7} \left(d_i \left(\frac{p_i - s_i - \alpha_i}{2}\right)\right) + \frac{1}{2}\sum_{i=0}^{7} w_i x_i + \frac{\beta_7}{2}$

Let $g_i = \frac{p_i + s_i - \beta_i}{2}$, $h_i = \frac{p_i - s_i - \alpha_i}{2}$

$$t_7^u = \sum_{i=0}^{7} d_i h_i + \sum_{i=0}^{7} \tilde{w}_i x_i + \sum_{i=0}^{7} g_i + \frac{\beta_7}{2}$$

weights × features bias.

d_i, x_i would be feature representations

$h_i, \tilde{w}_i$ are weights

This is the equation for the linear model to predict the time taken for the cipher slogged to reach the finish line. It might seem that the number of weights are $8+8 = 16$, 8 for h_i & 8 for $\tilde{w}_i$ $\left(\tilde{w}_i = \frac{w_i}{2}\right)$.

But for $i = 7$

$x_7 = d_7$

Therefore we can combine the weights in that case. So only 15 dimension.

$\therefore t_7^u = W^T \phi(c) + b$

where $W_{15\times 1} = \begin{cases} \frac{d_0}{2} & ; \text{if } i = 0. \\ \tilde{w}_i = \frac{w_i}{2} = \frac{d_i + \beta_{i-1}}{2} & ; \text{if } 1 \le i \le 6 \\ \frac{\beta_6 + p_7 - s_6}{2} & ; \text{if } i = 7 \end{cases}$

$$h_{(i-8)} = \frac{P_{i-8} - S_{i-8} - d_{i-8}}{2} \quad \text{if } 8 \le i \le 14$$

$$b = \sum_{i=0}^{7} g_i + P_7$$

$$\phi(c)_{15\times 1} = \begin{bmatrix} x_0 = d_0 d_1 \cdots d_7 \\ x_1 = d_1 d_2 \cdots d_7 \\ x_2 = d_7 \\ d_0 \\ d_1 \\ \vdots \\ d_6 \end{bmatrix}_{15\times 1}$$

$$\phi(c) = (\phi_i)_{15\times 1}$$

$$\phi_i = \begin{cases} x_i = d_i d_{i+1} \cdots d_7 & \text{if } 8 \le i \le 7 \\ d_{i-8} & \text{if } 8 \le i \le 14 \end{cases}$$

Hence we have proven that a Hebbian model can predict the Hamiltonian for a ~~sig~~ zephyr signal to reach the Jewish Lane.

Similarly, we can create 3 more models. Total of 4, one each for the zephyr & lower signals of POF$_0$ & POF$_1$

$$t^u_{POF_0} = w_a^T \phi(c) + b_a$$

$$t^u_{POF_1} = w_b^T \phi(c) + b_b$$

$$t^l_{POF_0} = w_c^T \phi(c) + b_c$$

$$t^l_{POF_1} = w_d^T \phi(c) + b_d$$

Response 0

$$\Delta_0(c) = L^{(c)}_{POF_0} - L^{(c)}_{POF_1}$$

$$= (W_c^T - W_d^T)\phi(c) + (b_c - b_d)$$

$$\Delta_0(c) = W_0^T \phi(c) + b_0$$

we need $\text{Response}_0 = \begin{cases} 0 & \text{if } \Delta_0(c) < 0 \\ 1 & \text{if } \Delta_0(c) > 0 \end{cases}$

$$\therefore \text{Response}_0 = \frac{1 + \text{sign}(W_0^T \phi(c) + b_0)}{2}$$

Response 1

Similarly,

$$\text{Response}_1 = \frac{1 + \text{sign}(W_1^T \phi(c) + b_1)}{2}$$

XOR

From the slides we know that the decision for XOR POF is

$$\frac{1 + (-1)^{n+1} \text{sign}\left(\prod_{i=1}^{n} (W_i^T \phi(c))\right)}{2}$$

From here on we will be including the bias term in the weights

Here $n = 2$

$$\therefore 1 - \frac{\text{sign}\left(W_0^T \phi(c)\, W_1^T \phi(c)\right)}{2}$$

We can convert

$$W_0^T \phi(c) \cdot W_1^T \phi(c) = W^T \Phi(c)$$

$\Phi(c)$ is created by multiplying each term of $\phi(c)$ with each other term.

A naive way of defining $\Phi(c)$ would be.

$$\Phi(c) = \begin{bmatrix} z_0 z_0 \\ z_0 z_1 \\ z_0 z_2 \\ \vdots \\ z_{15} z_{15} \end{bmatrix}_{256 \times 1} \quad \text{where } z_i \in \phi(c)$$

But this would have many repeated terms as $z_1 z_2$ and $z_2 z_1$ would be considered seperately.

We need to ~~find~~ find all possible combinations of picking 2 variables from a ~~pool~~ pool of 16 with repetition,

The formula is $^{n+r-1}C_r \Rightarrow {}^{16+2-1}C_2 = {}^{17}C_2 = 136$

$\therefore$ The dimension is 136