Arbiter PUF 逻辑回归建模

这里给出简单的建模方法，即对所有延迟设参数进行建模。

我们知道逻辑回归的公式是：

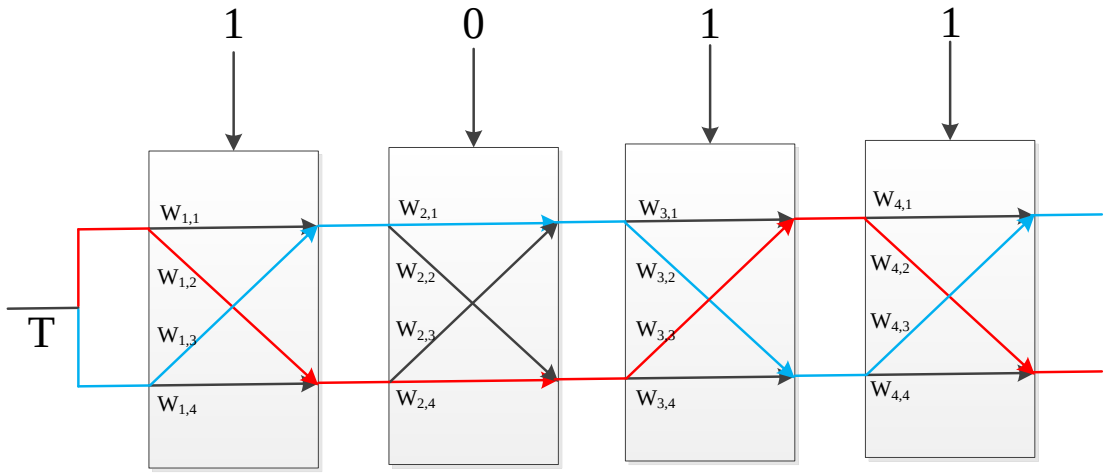
$$Y = g(w_0 + w_1x_1 + w_2x_2 + ... + w_nx_n)$$

相对于线性回归，它在多了 sigmoid 函数(上图公式中的 g()函数)，g()的作用是使得 Y 的值保持在 0 到 1 之间，其表达式如下：

$$g(z) = \frac{1}{1+e^{-z}}$$

我们拥有简单的逻辑回归公式，是不是简单的将 Arbiter PUF 的输入激励当做逻辑回归的输入$\{x_1, x_2, \ldots, x_n\}$就行了呢？显然是不行的。

我们建模要符合 Arbiter PUF 实际的工作原理。举如下例子：



如图，这是一个简单 4 阶 Arbiter PUF 对应激励(1011)的路径图，我们产生一位响应，只需要比较蓝红两条路径的信号传播快慢就行了。我们设逻辑回归的参数为{w0,w1,w2,w3,w4}，如果以(1011)作为输入，得到的结果为 w0+w1+w3+w4 显然没有任何意义。

这时就需要变通了，我们需要在输入或者设参数上做点手脚来遵循 Arbiter PUF 的工作原理。我这里给出一种最好理解的方式，即对 Arbiter PUF 的所有延迟段进行设参数。图中的$(w_{11}, w_{12}, w_{13}, w_{14})$对应之前图中的(p,s,t,q)，这时我们产生响应相当于比较(w12+w24+w33+w42)和(w13+w21+w32+w43)的大小。即判断以下两个矩阵对应位置相乘后求和的正负，

$$C = \begin{matrix} 0 & -1 & 0 & 0 \\ 1 & 0 & -1 & 1 \\ -1 & 0 & 1 & -1 \\ 0 & 1 & 0 & 0 \end{matrix} \qquad W = \begin{matrix} W_{11} & W_{21} & W_{31} & W_{41} \\ W_{12} & W_{22} & W_{32} & W_{42} \\ W_{13} & W_{23} & W_{33} & W_{43} \\ W_{14} & W_{24} & W_{34} & W_{44} \end{matrix}$$

可见激励(1011)和矩阵 C 是一一对应的，我们只需要对激励进行扩展就能完成建模。当然还有其他的设参数方式以及对应的激励扩展方式，但是万变不离其宗，建模还是要符合 Arbiter PUF 的工作原理，才能建出好的模型，大家加油。