Jinyang Ruan

011696096

CPT_S 515 Advanced Algorithms Mid-term

1. Step 1: Consider a walk ω with n nodes staring from v_0. Define the number of nondeterministic choices on this walk ω is $\#(\omega)$.

    The nondeterminism N on walk α shall be:

$$N = \frac{log_2 \#(\omega)}{n}$$

   Step 2: In the same way, I use P_n to denote the set of all walks in P with the length n. The total number of walks in P_n is $\#P_n$. A is the adjacency matrix of G. We have:

$$\#P_n = \det(A^n)$$

   Step 3: Modify the original graph G by separating the nondeterministic walks and the deterministic walks. Since every triggering event on each node holds different weights, we can consider the weight on each node is equivalent to the number of walk options on this node. After modification, we have new graph G', and the adjacency matrix of G' is A'. The total number of walks on G' is:

$$\#P'_n = \det\,(A'^n)$$

   Step 4: the nondeterminism M_1 on G shall be

$$M_1 = \limsup_{n\to\infty}\left(\frac{\log_2 \#P'_n}{n} - \frac{\log_2\ \#P_n}{n}\right)$$

   Then,

$$M_1 = \limsup_{n\to\infty}\left(\frac{\log_2 \det\,(A'^n)}{n} - \frac{\log_2 \det\,(A^n)}{n}\right)$$

Step 5: Since matrices A' and A are positive. We can use Perron-Frobenius theorem to get the largest eigenvalue of A'^n and A^n when $n \to \infty$, which could be represented as follow:

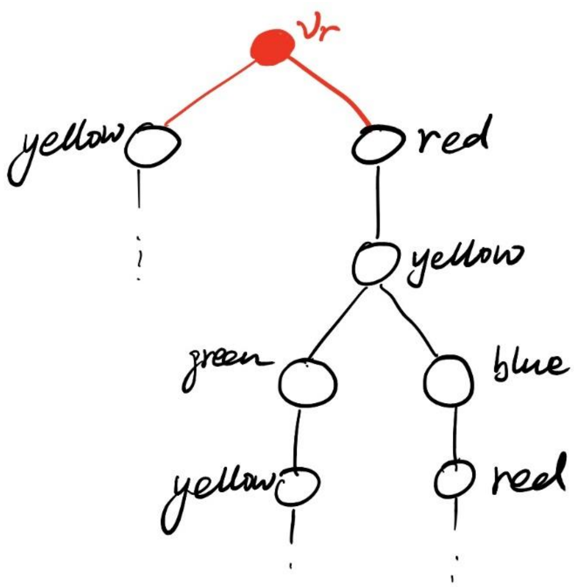
$$\lambda = \limsup_{n\to\infty} \left(\frac{\log_2 \det (A^n)}{n}\right)$$

Step 6: So far, M_1 can be represented by:

$$M_1 = \lambda' - \lambda$$

2. Step 1: Since we have all color sequences in C, and all the walks represented by color sequences starting from v_0, I add a color Red before all color sequences in order to let this sequences have same starting value while the nondeterminism will not change.

   Step 2: Build a new graph G' with the starting node v_r which takes Red property. Each color sequence will be represented as a walk starting from v_r in G'. Same prefix color sequences shared the same walk in G'.



So far, the nondeterminism on all color sequences in C is equivalent to the average brunches on each node in G'.

Step 3: Set A as the adjacency matrix of G'. The total number of branches with the length n in G' is $\#P_n$,

$$\#P_n = \det(A^n)$$

Step 4: the nondeterminism M_2 on graph G' shall be represented as:

$$M_2 = \limsup_{n\to\infty}\left(\frac{\log_2 \det(A^n)}{n}\right)$$

Step 5: According to the Perron-Frobenius theorem, we can get the largest eigenvalue λ of A^n when $n \to \infty$,

$$\lambda = \limsup_{n\to\infty}\left(\frac{\log_2 \det(A^n)}{n}\right)$$

Step 6: The nondeterminism M_2 on all color sequences in C is,

$$M_2 = \lambda$$

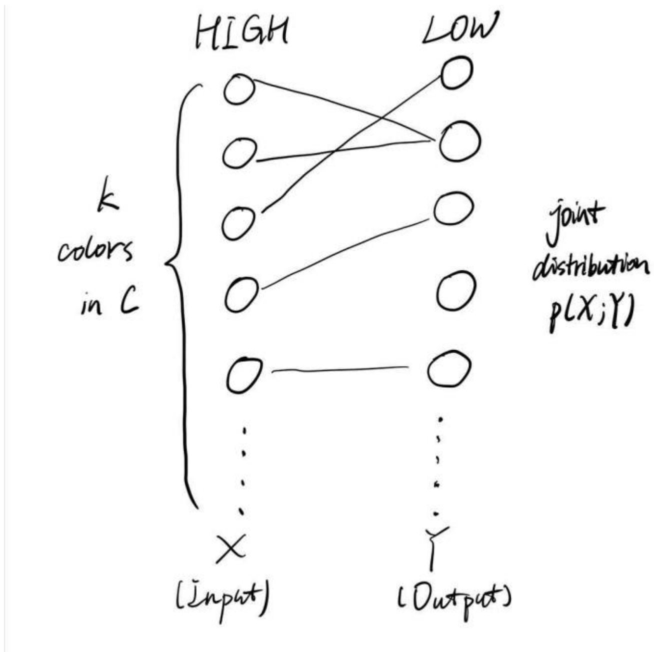
3. Translation: The nondeterminism of the Blackbox program under test is also the uncertainty of the program. I suppose that the nondeterminism of the program under one test is also the "information" of the program under one test. So, I am going to compute how much information leaked from inputs to outputs under one test.

Step 1: Since inputs and outputs are dependent, I consider two dependent variables X, Y

$$X =< i_1, i_2, \dots, i_n >$$

$$Y =< o_1, o_2, \dots, o_n >$$

Step 2: Make a bipartite graph G' with joint distribution $p(X;Y)$, Assume that the number of finite colors in C is k.



Step 3: Use $I(X;Y)$ to denote Shared information/entropy between variables X and Y, from $p(X;Y)$, we can compute $I(X;Y)$. We define

$$\lambda_{G'} = \max_{p(X;Y)} I(X;Y)$$

Step 4: If I can compute the size of max matching M, then

$$\lambda_{G'} = \log_2 M$$

However, since k is very large and n tends to infinity,

$$\lambda_{G'} = \limsup_{n\to\infty}(\frac{1}{n}\log_2\frac{N_{Left}\cdot N_{Right}}{E})$$

Then, we have the nondeterministic of the Blackbox program under one test result.

$$\lambda_{G'} = \limsup_{n\to\infty}(\frac{1}{n}\log_2\frac{k^2}{n})$$

Step 5: Since there are multiple test results. We have multiple λ_x $(x > 0\ and\ x \to \infty)$ for each test results. Compute the variance S^2,

$$S^2 = \frac{\sum_{i=1}^{x}(\lambda_i - avg.\lambda)}{x} \quad (x > 0\ and\ x \to \infty)$$

If the variance S^2 is big, I can consider the nondeterminism of Blackbox program under test is high, otherwise, nondeterminism of Blackbox program under test is low.