

Given below are examples to show how the two algorithms work in the particular case of 3 genes:

Example 3.1 (Algorithm 1)

Suppose that $k = 2, m = 1, M = 2, l = 1,$ and $L = 5.$

Next, suppose that the states 000 and 011 were generated by STEP1.

STEP2: Suppose W is generated where $W_1 = \{x_2, x_3\}, W_2 = \{x_1, x_3\}, W_3 = \{x_1, x_2\}$

STEP3: Table 1 shows that the attractors generated in STEP1 are compatible with $W.$ The remaining entries a_1, \dots, a_6 in the truth table are filled in randomly in the next step of the algorithm. One can notice certain patterns in the entries in each one of the three columns of the table. That reflects the structure of the predictor set $W,$ and reduces the number of the possible ways to randomly fill in the missing entries during the next step of the algorithm.

On the other hand, if the attractors generated in STEP1 were 000 and 100, then we must have for the predictor function of the first gene $x_1: f_1(0, 0) = 0,$ while from the second attractor, we get $f_1(0, 0) = 1$ which is a contradiction. Therefore that attractor set is not compatible with the W generated in STEP2.

STEP4: Here we randomly fill in the remaining entries of the truth table. Suppose that this produces $a_1 = 0, a_2 = 0, a_3 = 1, a_4 = 0, a_5 = 0, a_6 = 1.$ This selection produces the following transitions in the state transition diagram: $0 \rightarrow 0; 1 \rightarrow 2; 2 \rightarrow 1; 3 \rightarrow 3; 4 \rightarrow 2; 5 \rightarrow 0; 6 \rightarrow 3; 7 \rightarrow 1.$ Here we used the decimal representation of the states. It is clear that during the STEP5 of the algorithm the cycle $1 \rightarrow 2; 2 \rightarrow 1$ will be discovered, which will cause the BN generated by the present truth table to be discarded, and we will be returned to STEP4.

On the other hand if we had $a_1 = 0, a_2 = 1, a_3 = 1, a_4 = 0, a_5 = 0, a_6 = 1$ produced in STEP4 then the transitions in the state transition diagram would be $0 \rightarrow 0; 1 \rightarrow 2; 2 \rightarrow 5; 3 \rightarrow 3; 4 \rightarrow 2; 5 \rightarrow 0; 6 \rightarrow 7; 7 \rightarrow 1.$ Since the only cycles here are those within the attractor set, STEP5 of the algorithm will take us to STEP6. STEP6 will detect that there are 5 level sets, and this will take us to STEP7.

x_1	x_2	x_3	f_1	f_2	f_3
0	0	0	0	0	0
0	0	1	a_1	1	0
0	1	0	a_2	0	1
0	1	1	0	1	1
1	0	0	0	a_3	a_4
1	0	1	a_1	a_5	a_4
1	1	0	a_2	a_3	a_6
1	1	1	0	a_5	a_6

Table 1: Truth Table for Example 3.1

Example 3.2 (Algorithm 2)

Suppose that $k = 2, m = 1, M = 2, l = 1$ and $L = 3.$

Next, suppose that the transition diagram shown in Figure 1 was randomly generated in STEP1. The truth table resulting from STEP2 is shown in Table 2. STEP3: It is clear from this truth table that $W_1 = \{x_1, x_2, x_3\},$ and since it has more than $M = 2$ elements the algorithm returns to STEP1.

On the other hand if the transition diagram shown in Figure 2 was generated in STEP1, then STEP2 would produce the truth table shown in Table 3.

Now each $W_i; i = 1, 2, 3$ has no more than $m = 2$ elements, and the algorithm successfully terminates producing a BN with the truth table shown in Table 4 and state transition diagram from Figure 2.

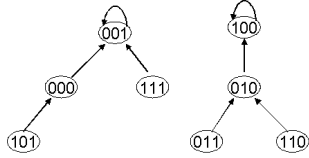


Figure 1: First State Transition Diagram for Example 3.2

Gene Values	f_1	f_2	f_3
0 0 0	0	0	1
0 0 1	0	0	1
0 1 0	1	0	0
0 1 1	0	1	0
1 0 0	1	0	0
1 0 1	0	0	0
1 1 0	0	1	0
1 1 1	0	0	1

Table 2: First Truth Table for Example 3.2

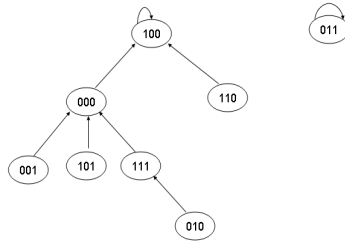


Figure 2: Second State Transition Diagram for Example 3.2

Gene Values	f_1	f_2	f_3
0 0 0	1	0	0
0 0 1	0	0	0
0 1 0	1	1	1
0 1 1	0	1	1
1 0 0	1	0	0
1 0 1	0	0	0
1 1 0	1	0	0
1 1 1	0	0	0

Table 3: Second Truth Table for Example 3.2