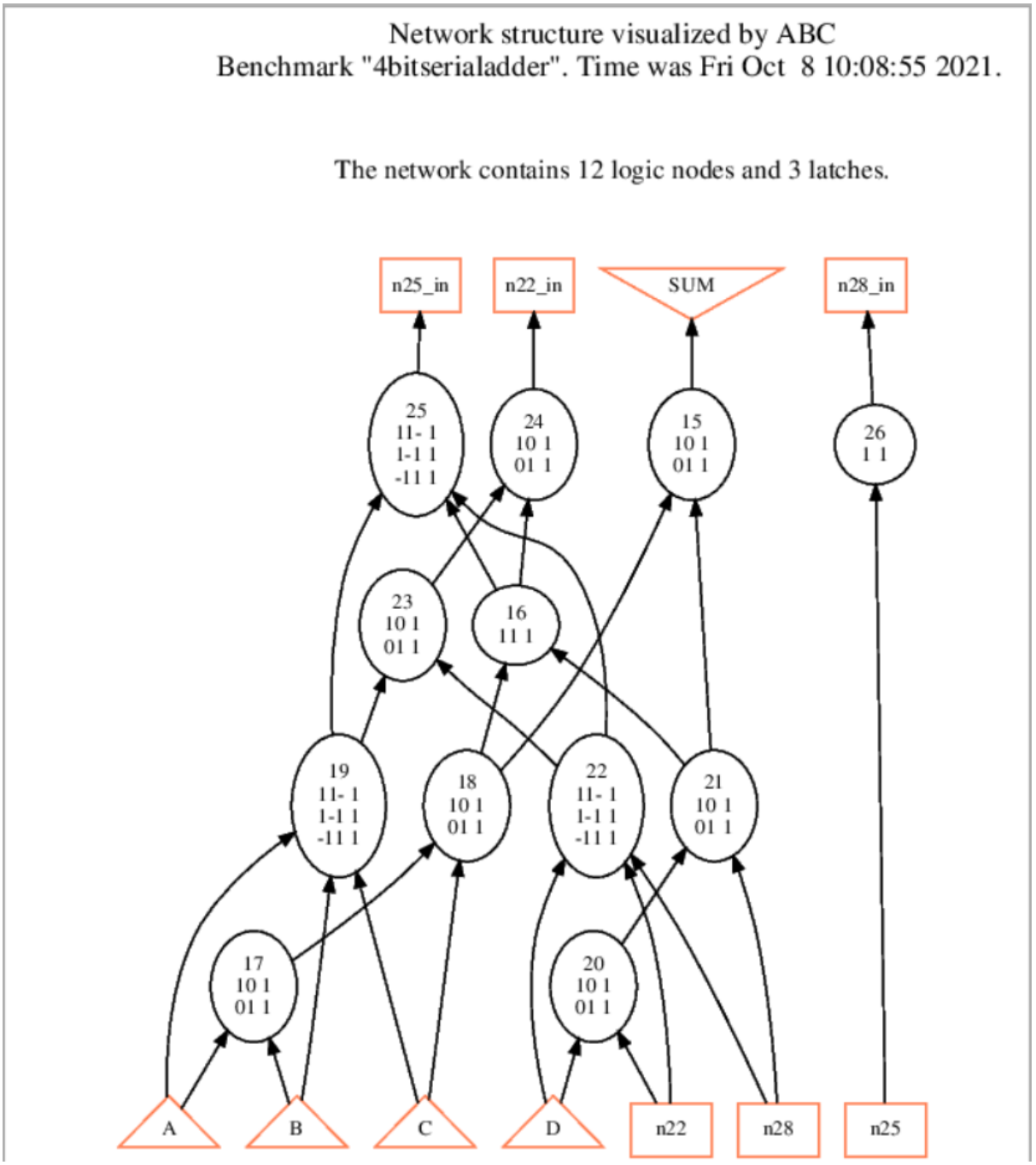
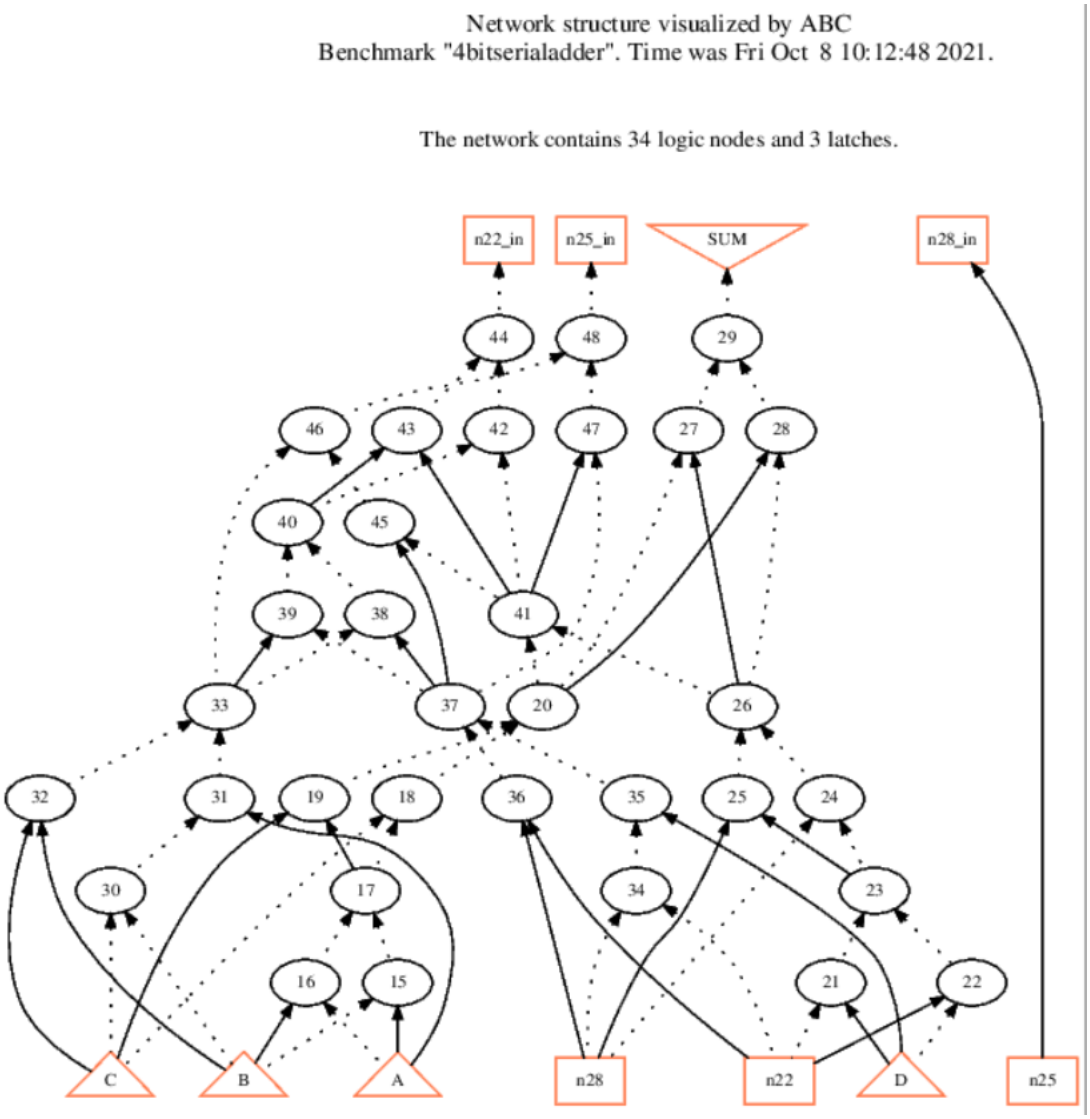
1. 一頁一張

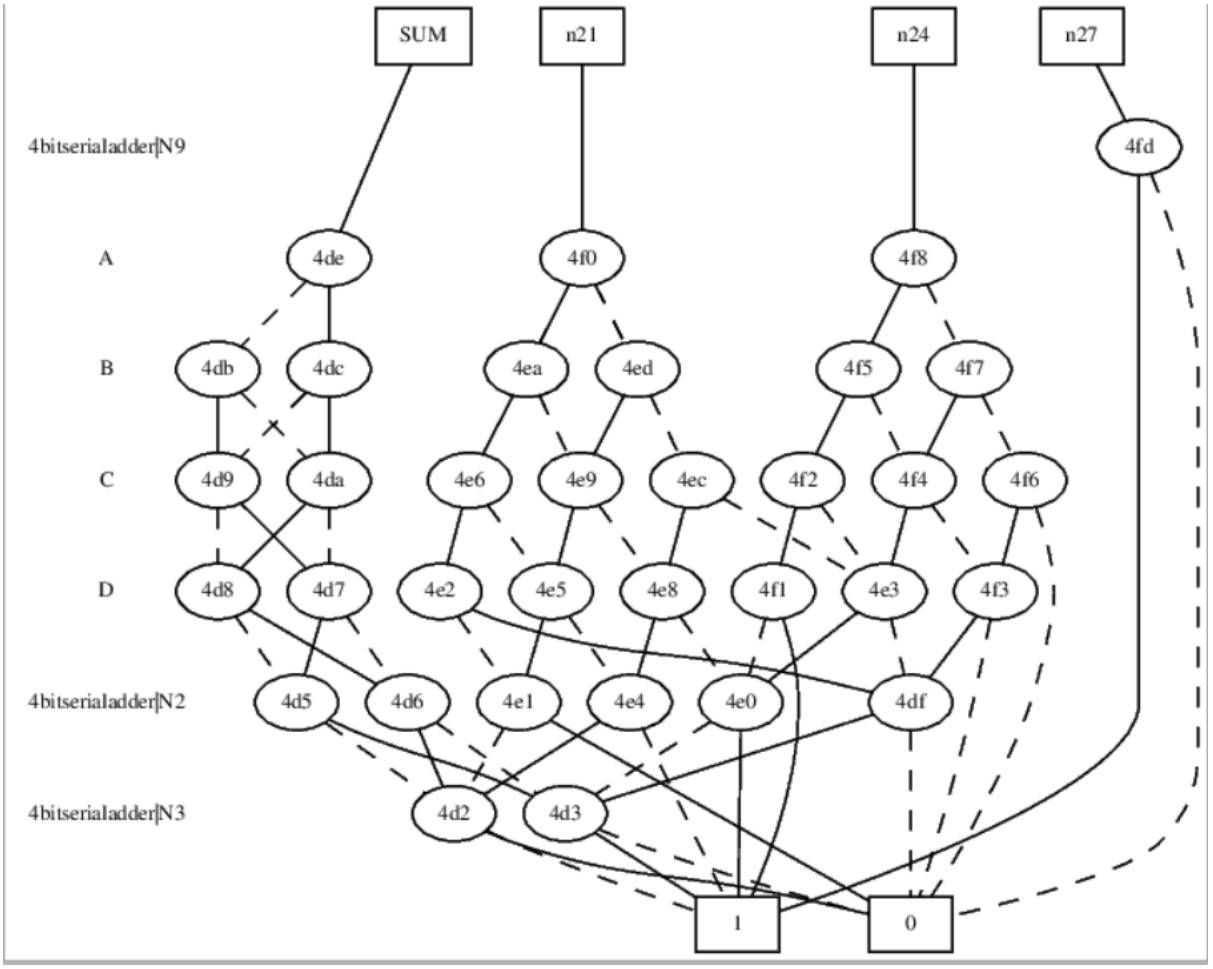
visualize the network structure：



visualize the AIG：



visualize the BDD：



2.

(a) Compare the following differences with the four-bit adder example.

(1) logic network in AIG (by command `aig`) vs. structurally hashed AIG (by command `strash`)

```
abc 02> aig
abc 02> print_stats
4bitserialadder             : i/o =    4/    1  lat =    3  nd =    12  edge =    26  aig  =    34  lev = 4
```

aig 的功能是將原本的 logic network 裡面的 gates 都變成 and-inv 的形式

```
abc 02> strash
abc 03> show
abc 03> Warning: Missing charsets in String to FontSet conversion

abc 03> print_stats
4bitserialadder               : i/o =    4/    1  lat =    3  and =     34  lev =  8
```

strash的功能則是直接用 AIG graph 代表整個 logic network

雖然兩者的功能都是把 logic network 化成 AIG，但 aig 應該只是把原本的SOP給拆開來(所以依然還是SOP的狀態，改變的只有node function)，但 strash 則是把原本的 logic network 轉換成由 POs 連到 PIs 的 AIG

(2) logic network in BDD (by command `bdd`) vs. collapsed BDD (by command `collapse`)

```
abc 02> bdd
abc 02> show
abc 02> Warning: Missing charsets in String to FontSet conversion

abc 02> print_stats
4bitserialadder             : i/o =    4/    1  lat =    3  nd =    12  edge =    26  bdd  =    28  lev = 4
abc 02>
abc 02>
abc 02> collapse
abc 03> print_stats
4bitserialadder             : i/o =    4/    1  lat =    3  nd =     4  edge =    19  bdd  =    28  lev = 1
```

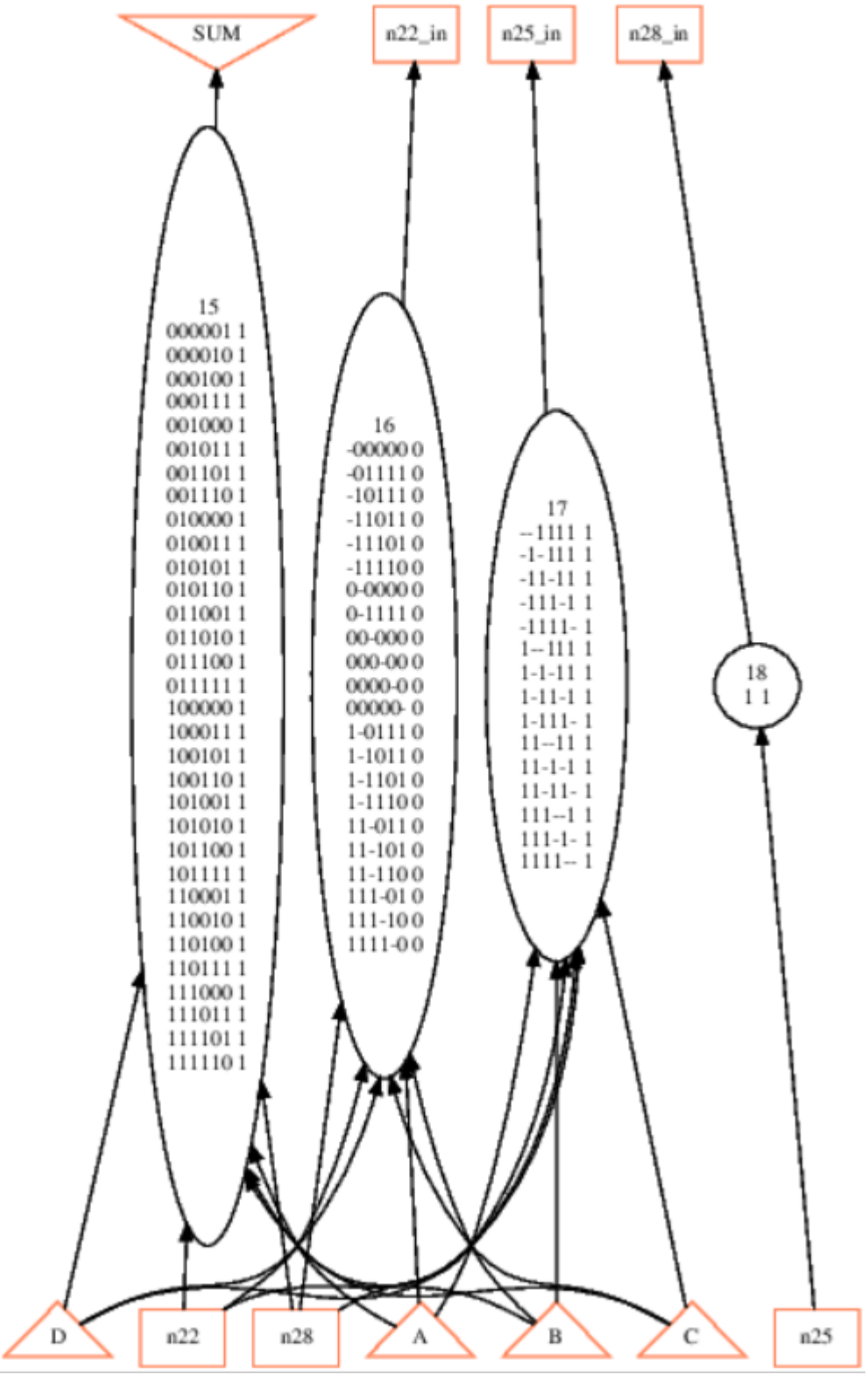
bdd 的功能是將 logic network 用 BDD 的表示方式存起來

collapse的功能是把整個logic network改成用一個 level 表示

下頁的圖可以看到collapse之後再用show而非show_bdd的結果，可以發現它會把整個logic network變成一個SOP

Network structure visualized by ABC
Benchmark "4bitserialadder". Time was Fri Oct 8 10:40:49 2021.

The network contains 4 logic nodes and 3 latches.



(b) Given a structurally hashed AIG, find a sequence of ABC command(s) to covert it to a logic network with node function expressed in sum-of-products (SOP).

>logic (就一個)