On Coloring and Colorability Analysis of Integrated Circuits with Triple and Quadruple Patterning Techniques

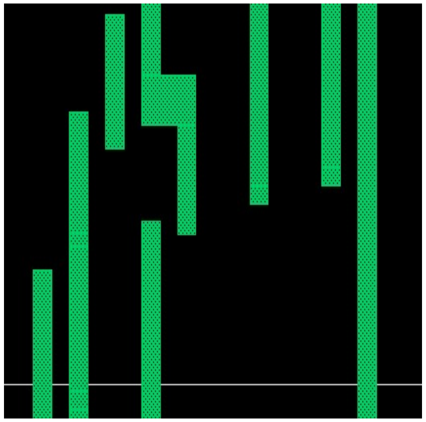
Alexey Lvov

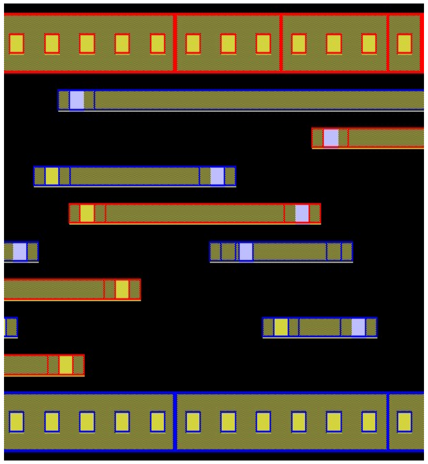
Gus Tellez

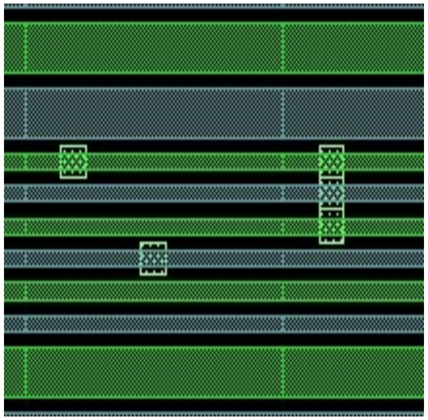
Gi-Joon Nam

Background and motivation

- Manaufacturing difficulty
  - 22nm:
    - Multi directional single patterned
  - 14nm:
    - Uni directional double patterning
  - 7nm:
    - Uni directional Self Aligned Double Patterning







- Coloring decomposition & stitching
  - Mostly Post-fix heuristic methods
  - Lack of theoretical / constructive approaches

Background: Multi-patterning and coloring problem

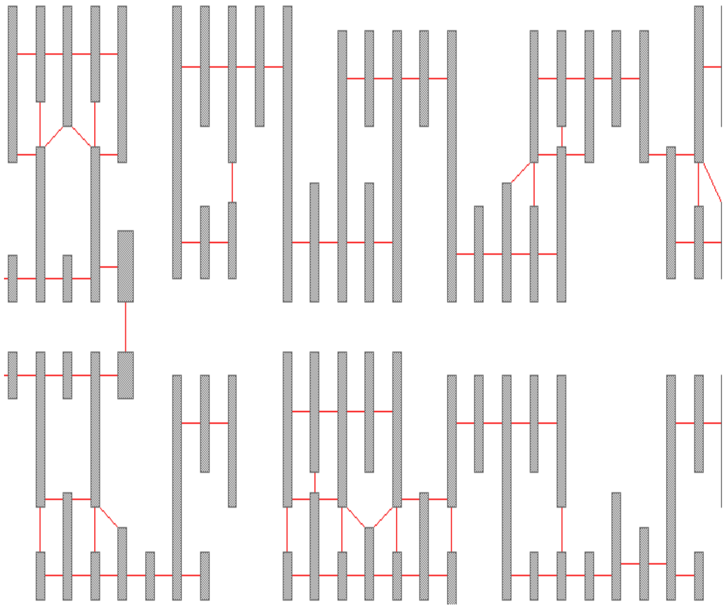
- In Litho-Etch-Litho-Etch multi-patterning, shapes on one layer are assigned to a unique mask.
- The layer is subdivided into k masks (k = 2, 3 or 4).
- Two rules exist:
  1. Two shapes that belong to one and the same mask must have a large minimum distance between them.
  2. In order to legally place two shapes at a very small separation distance place them on two different masks.
- Given a layout, the masks can be determined by solving a k-coloring problem on a constraint graph, where:
  - The color represents a mask
  - Nodes represent shapes
  - Arcs occur when shapes are at a distance less than the SAME MASK spacing rule
  - This graph is called a "conflict graph"

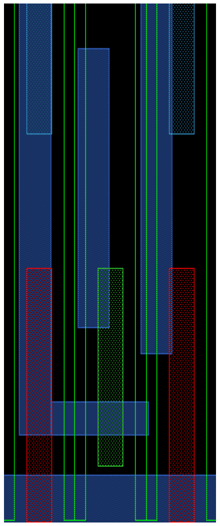
Layer coloring model & algorithm

Some of 7nm layers require FLAT 4-coloring

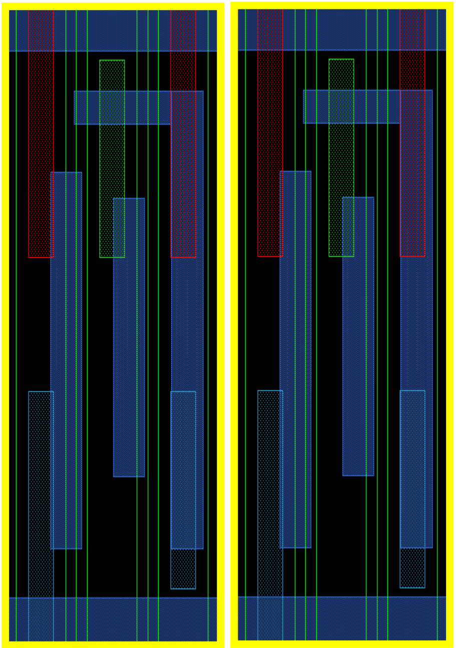
- Coloring CANNOT be done at a standard cell level
- Coloring must be done post-placement

- Example of uncolored layer layout





Shapes occur at critical distances across cell rows

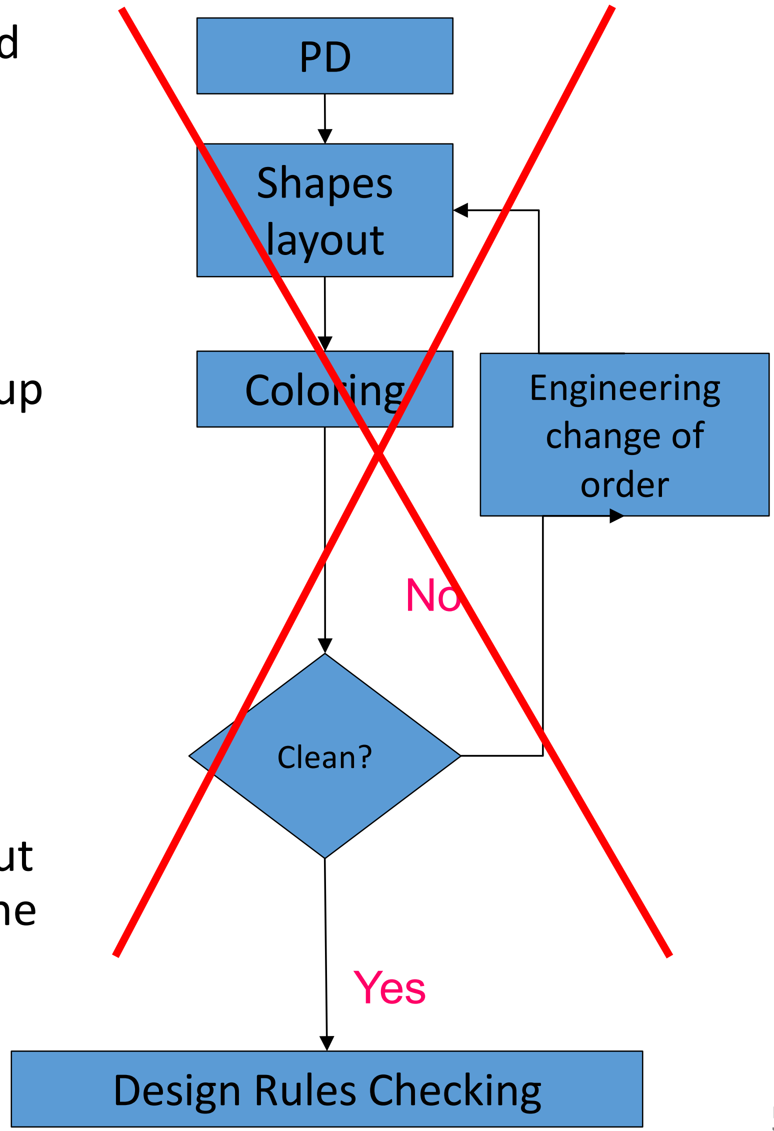


Shapes touch across cell rows

Flat layer 4 coloring possibilities

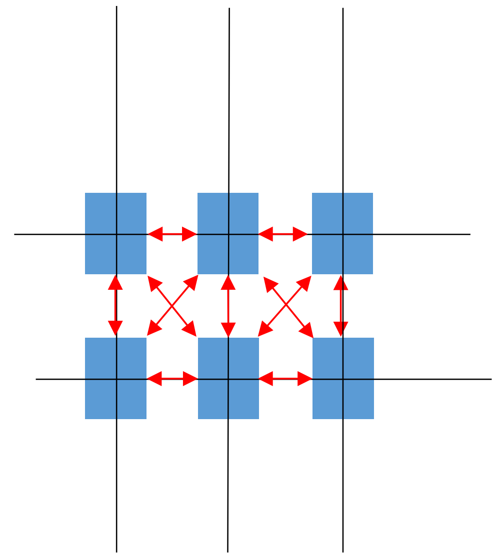
- 4-coloring problem is NP-complete
  - But are the coloring instances we will see hard to color?
- FLAT coloring means that coloring must be done:
  - During placement : run time prohibitive.
  - Post placement : can lead to slow manual fix up loop.
  - How likely are placements that cannot be colored?
    - No quantification method yet.
- Our answer:
  - Correct-by-construction method: Build a layout model which set of design rules guarantees the colorability of a layout.

Post-processing method



Layout Model

- Blue rectangles are layout shapes.
- Unit width shapes.
- Shapes occur on a grid.
- Conflict graph:
  - For every shape, there is a node in the graph.
  - There is an arc between nodes, when the corresponding shapes occur at or less than a minimum distance.



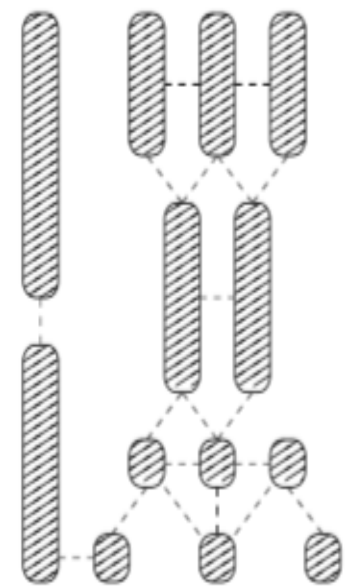


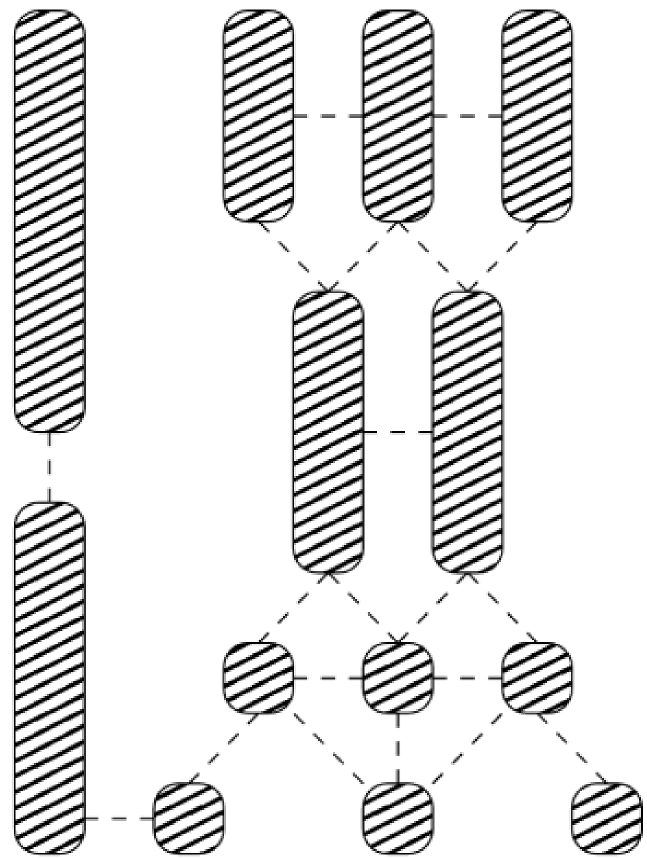
Fig. 1. Model A. Dashed lines show adjacency.

A *model* is a set of all layouts that follow specified design rules together with a definition of adjacency of a pair of shapes of a layout.

Layout models & constructive coloring algorithms with 4 colors

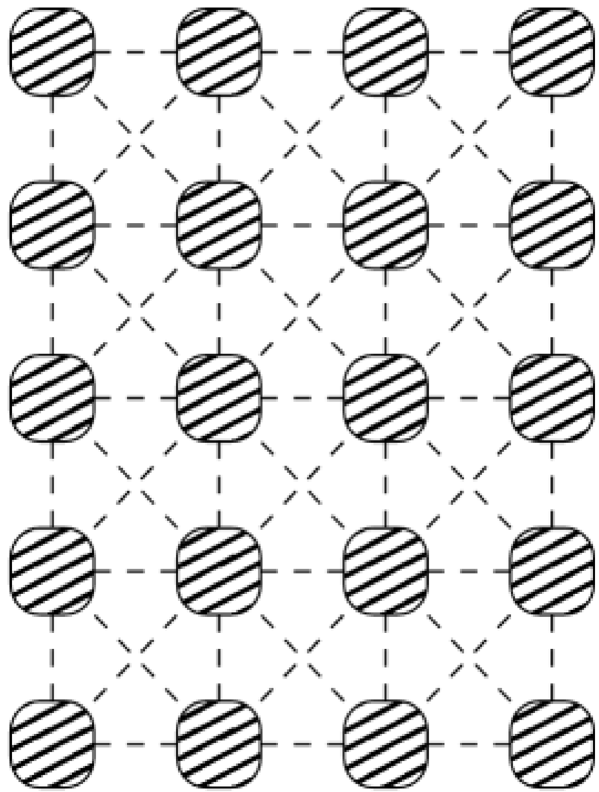
- Model A
  - A layout model that shows most standard cell placements are easily colorable
- Model B
  - A layout model incorporating the “cross-couple” with model A, where layouts are easily colorable
- Models C and D
  - Layout models that shows that a small addition of flexibility in model A can lead to un-colorable layouts

Model A: Vertical shapes only.



Dashed lines show adjacency.

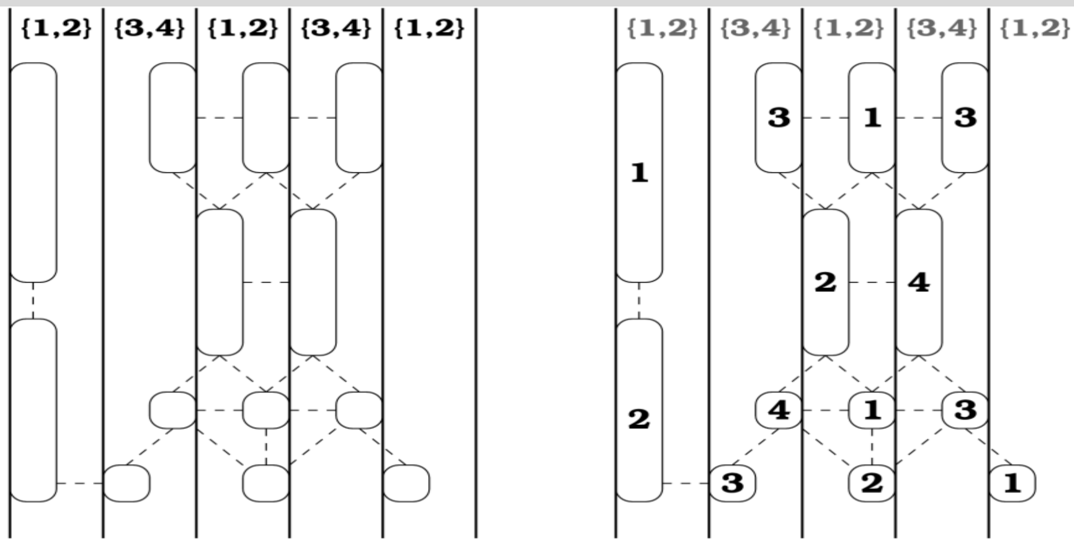
Model A: Non-planar example.
E = 55,   3 * V – 6 = 54.



Can not apply Four Color Map theorem.

Model A: Coloring Algorithm.

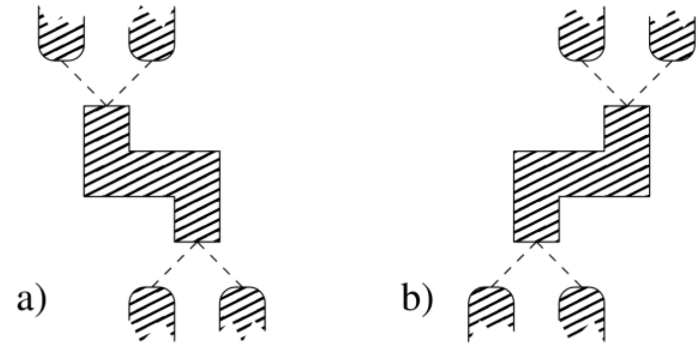
1. Split layout into columns of width 2.
2. Split the set of available four colors into two pairs: $\{1, 2\}$ and $\{3, 4\}$.
3. Starting with the first column, label odd columns with $\{1, 2\}$ and even columns with $\{3, 4\}$. See Figure 3a.
4. In each column we will use only the colors from its label.
5. By the design rules shapes are at least one square apart so they are linearly ordered in each column from top to bottom.
6. Color the shapes in each column from top to bottom using the two colors from the columns label in alternating order. See Figure 3b.

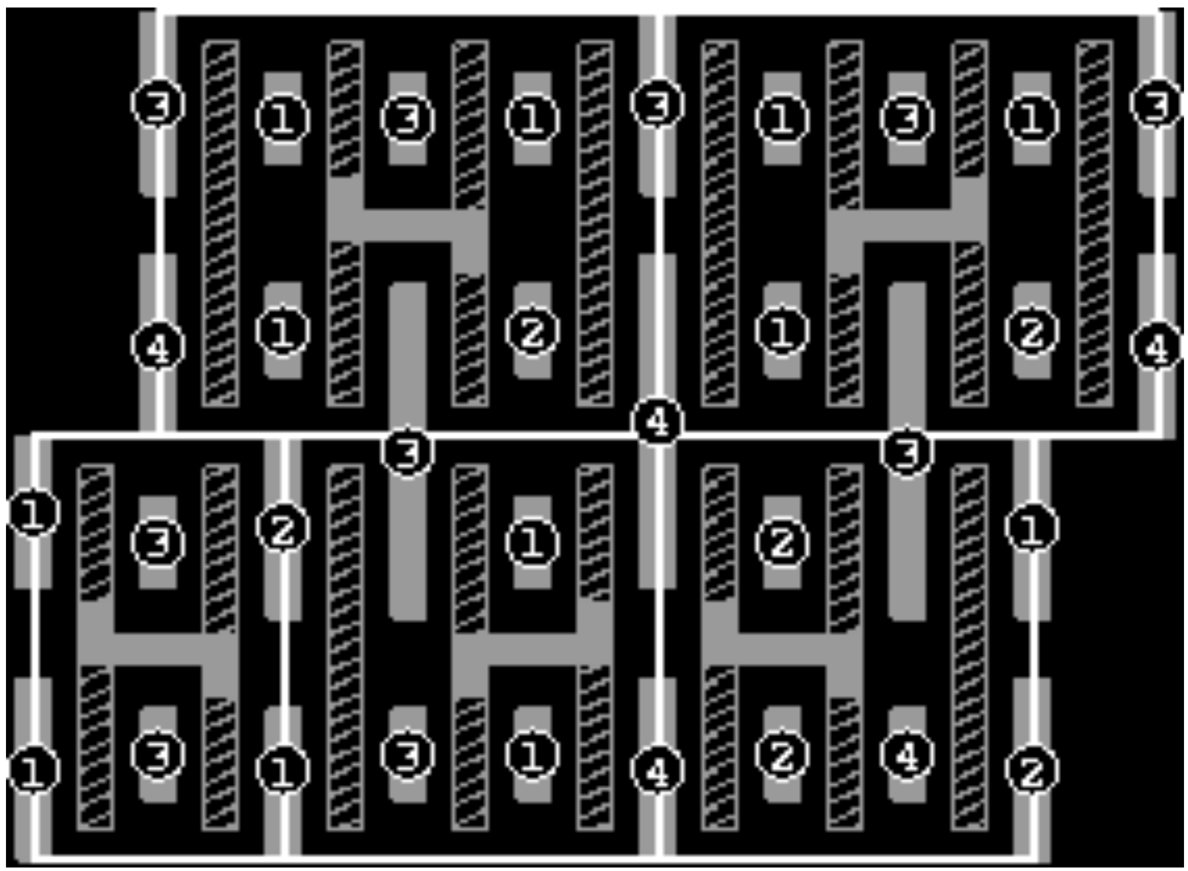


a b

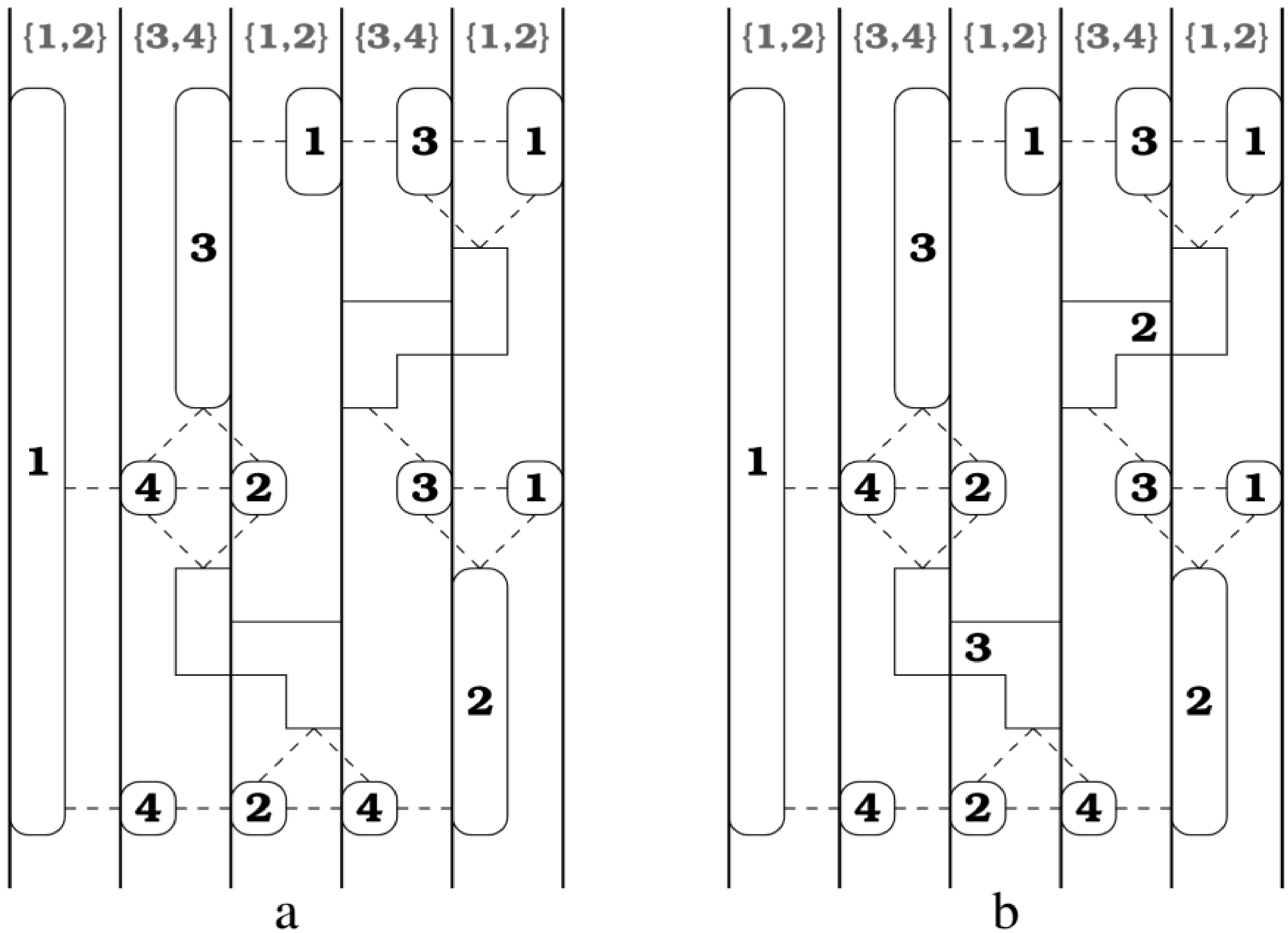
Model B:

Cross-link shapes are used in many standard cells. For example In XOR cell.





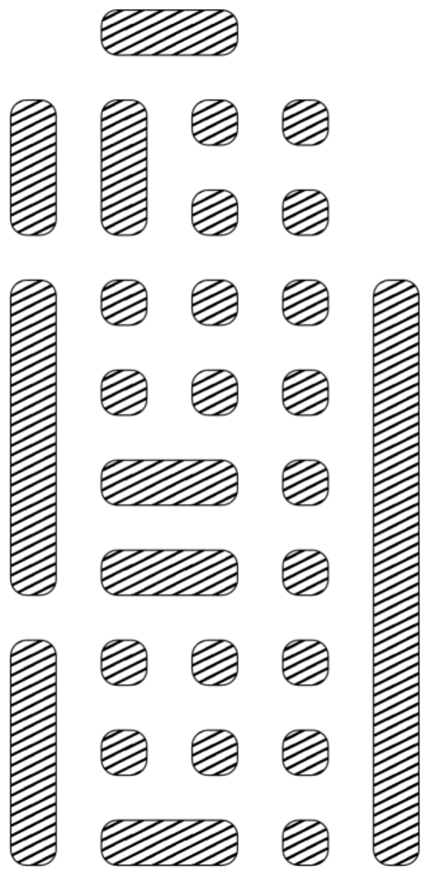
Model B: Coloring Algorithm



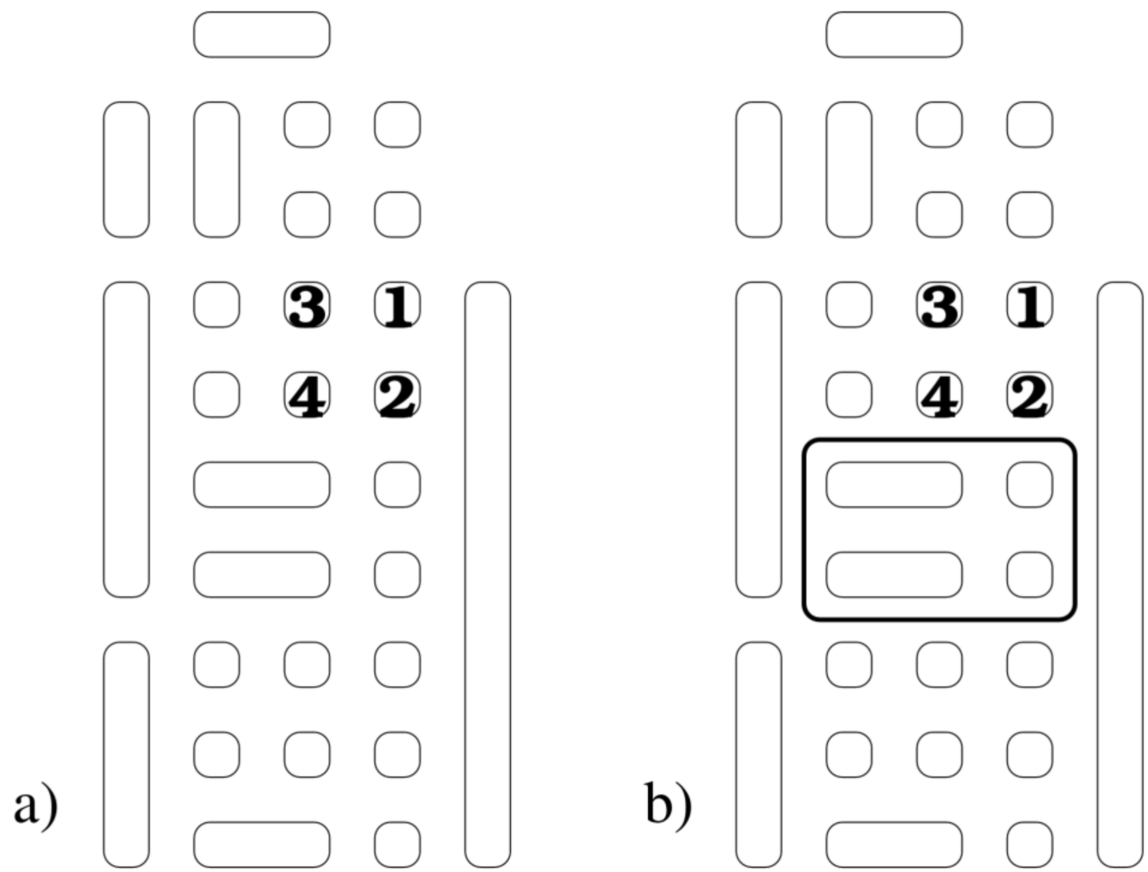
Once we have a four colorable model it is a natural question to ask by how much can we relax the design rules without breaking the four colorability property. In this section we show that both models A and B are almost tight in that sense

NOT four colorable models.

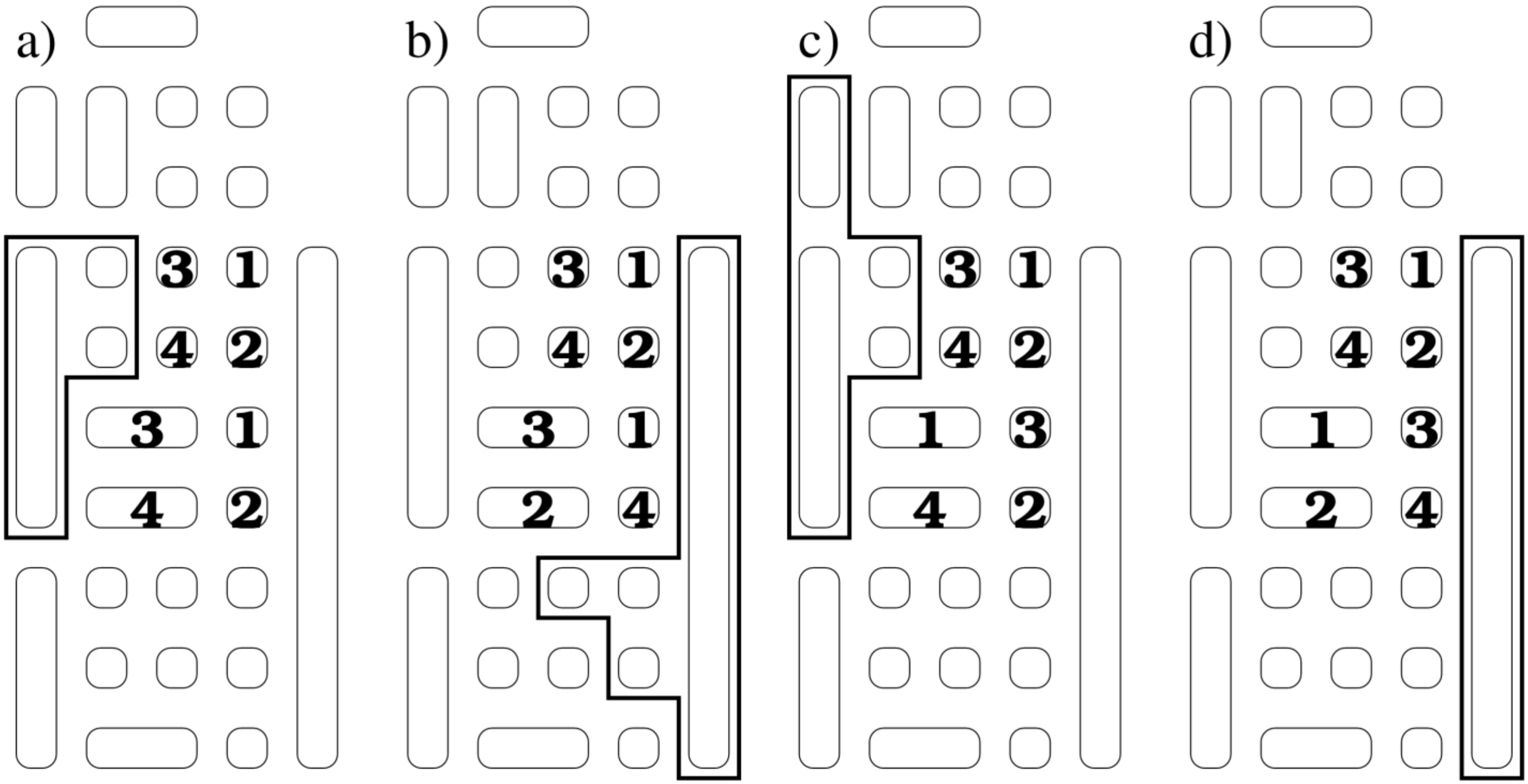
Model C: Take model A and allow horizontal shapes of size 3-by-1.



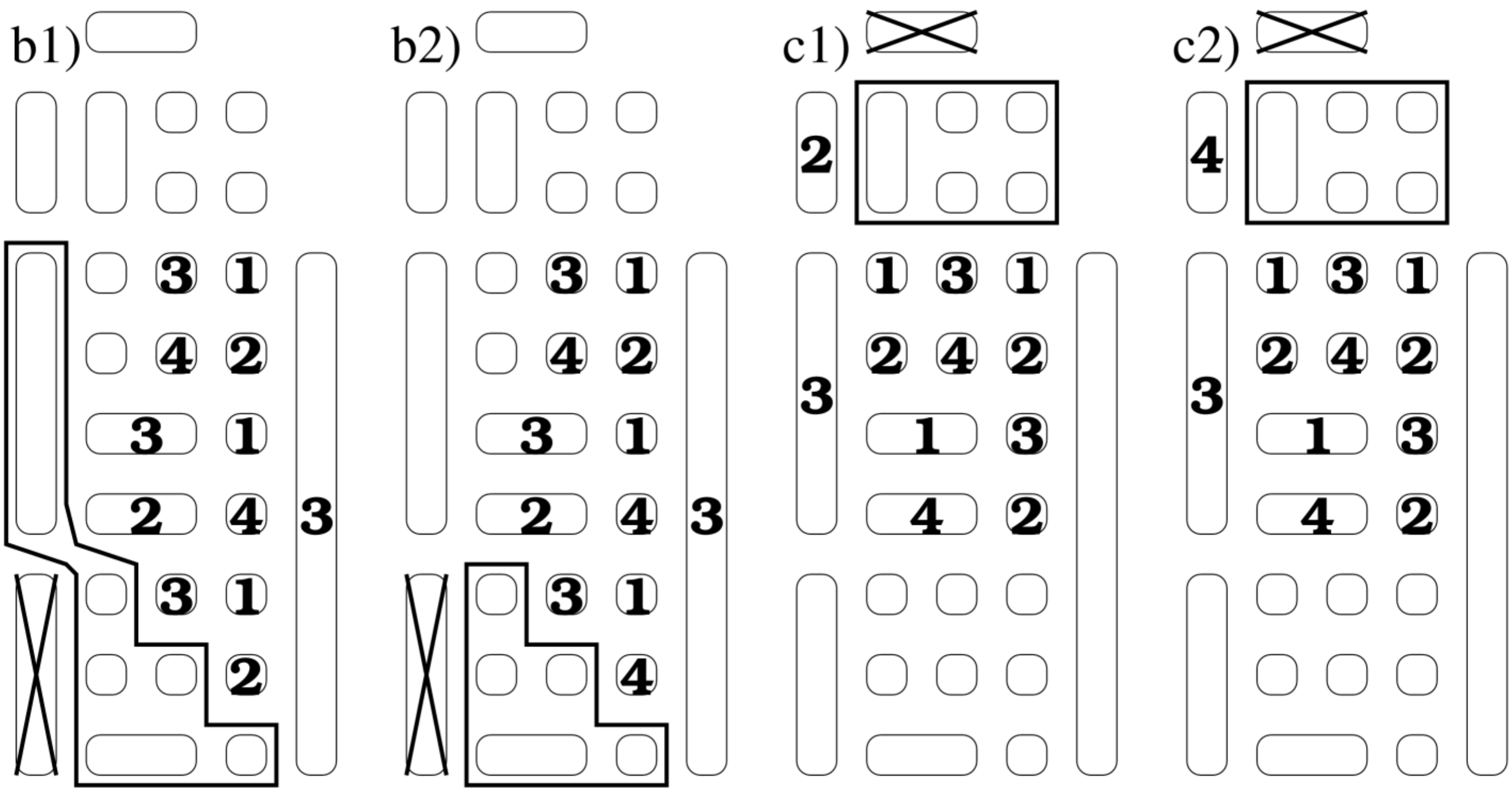
Model C is not 4-colorable. Proof (1 of 3)



Model C is not 4-colorable. Proof (2 of 3)

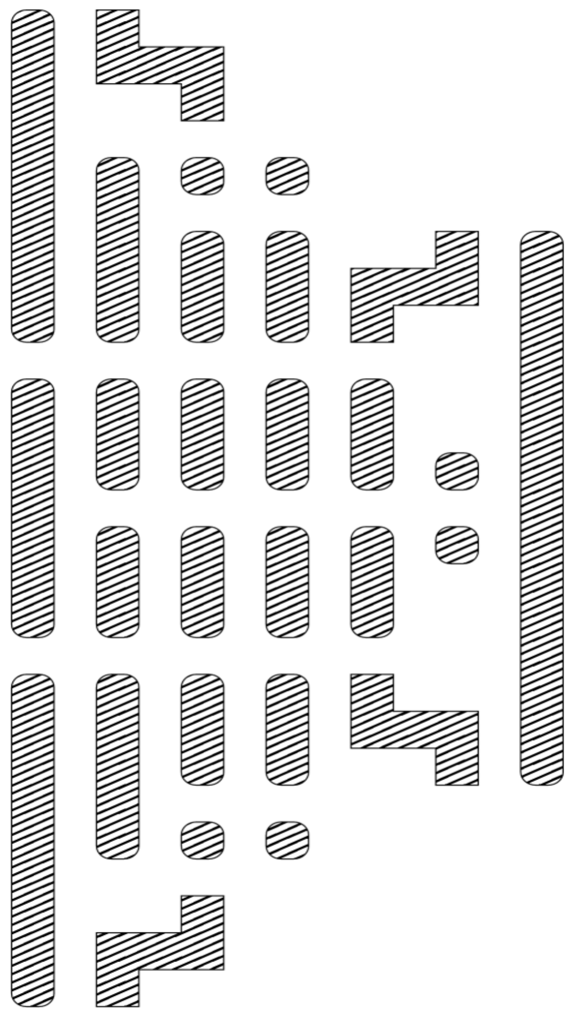


Model C is not 4-colorable. Proof (3 of 3)

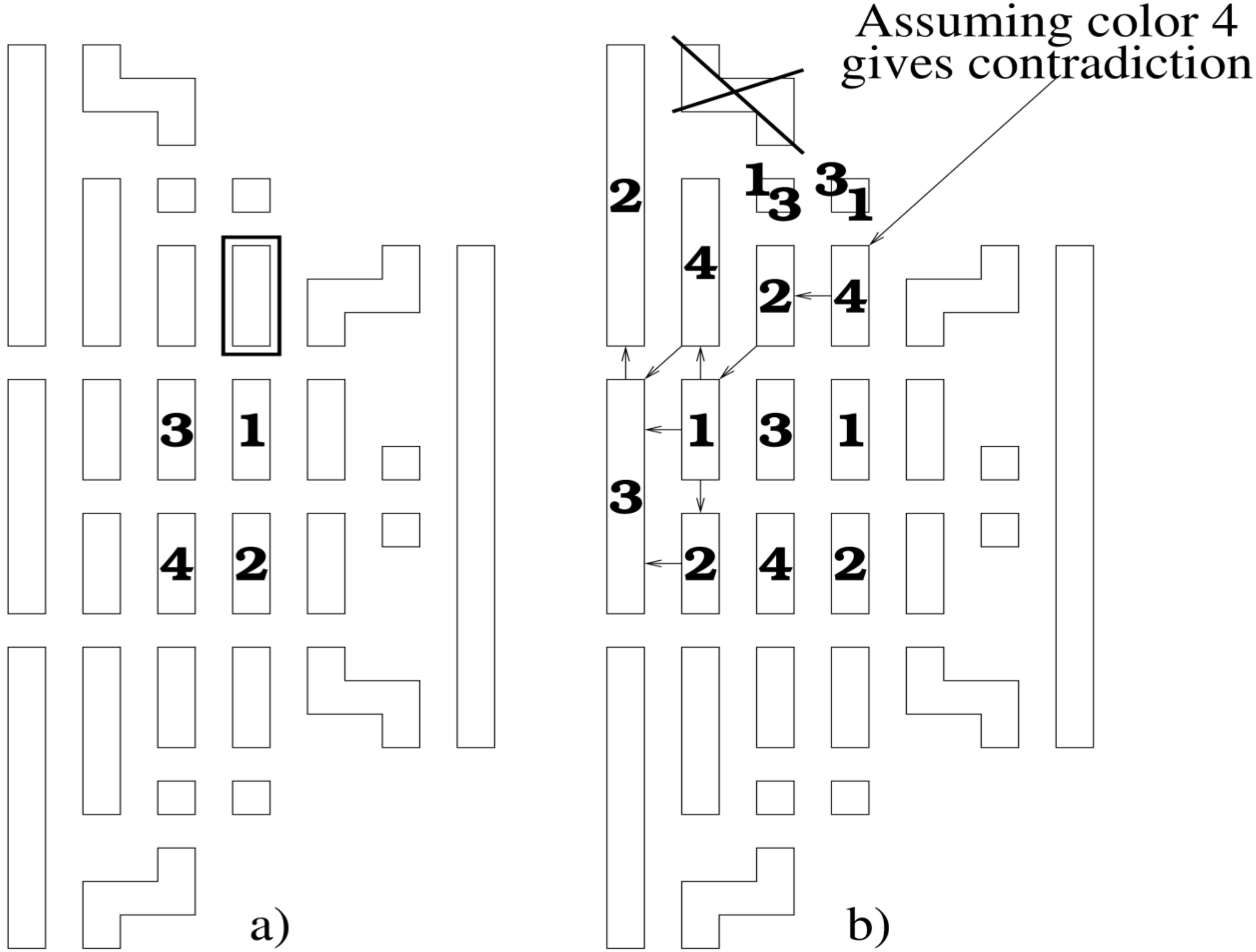


NOT four colorable models.

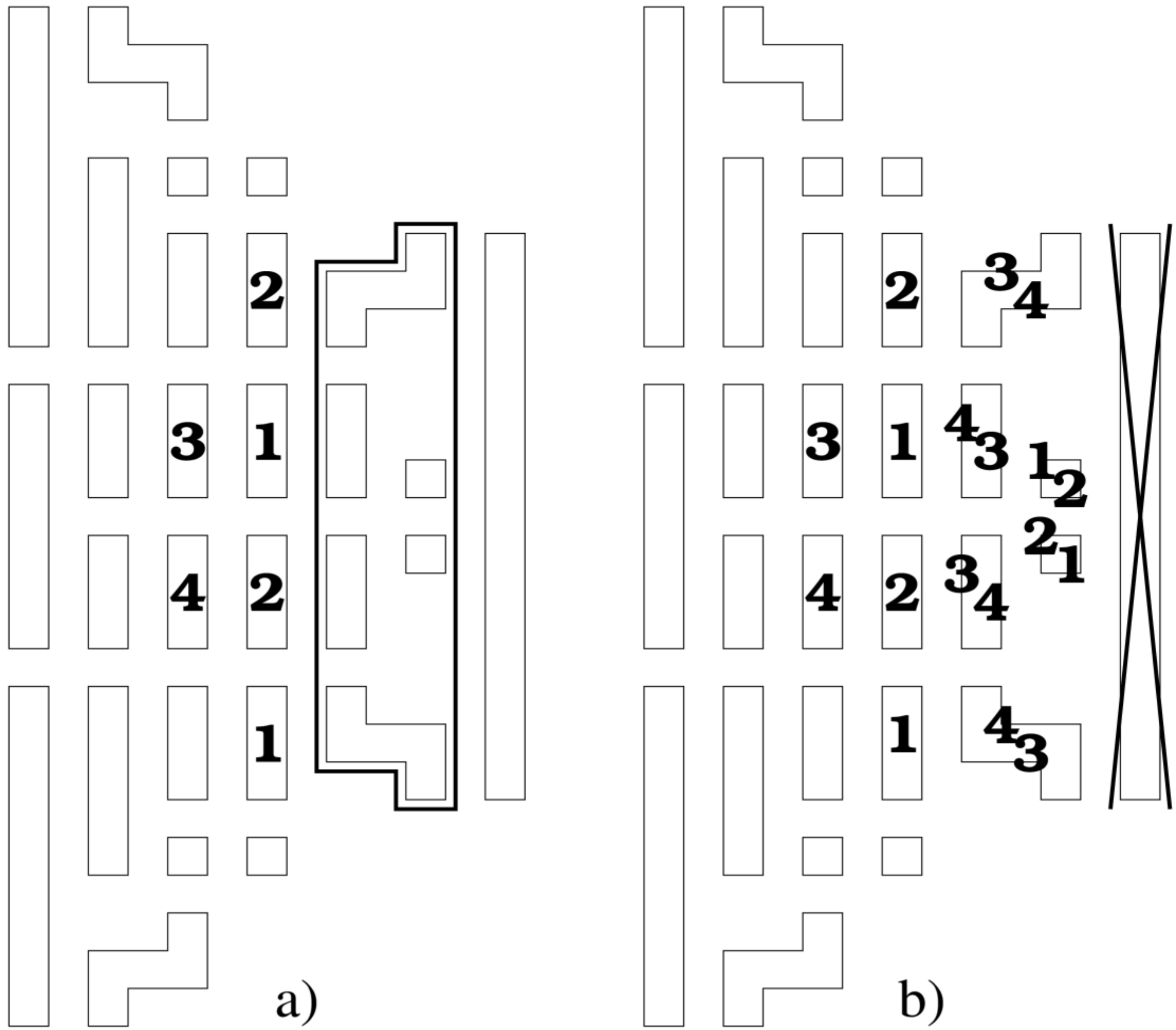
Model D: (A very slightly relaxed model B)



Model C is not 4-colorable. Proof (1 of 2)

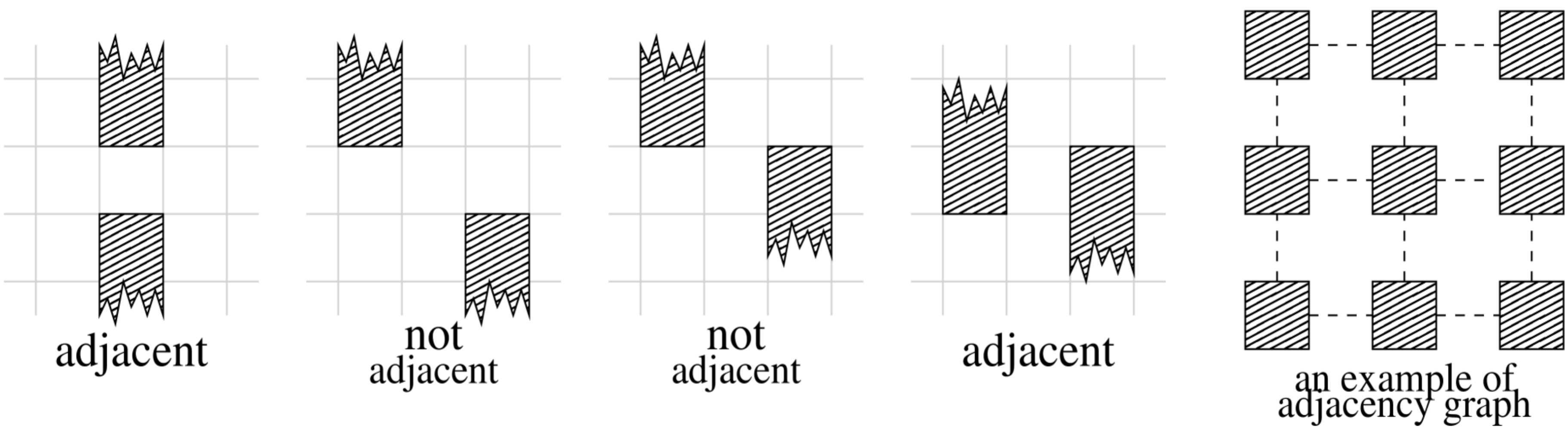


Model C is not 4-colorable. Proof (2 of 2)

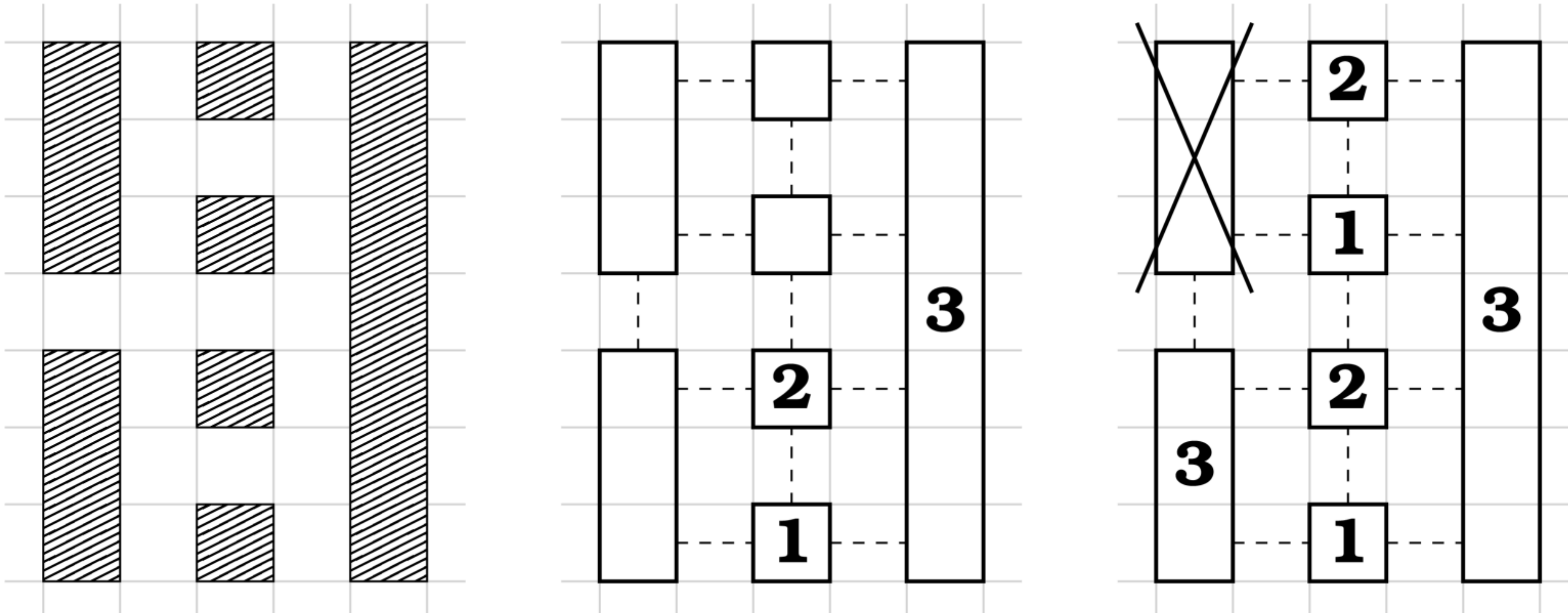


Three colorability. Model E.

• All shapes are vertical rectangles of width one separated by space of width at least one, which vertexes have integer coordinates and which vertical center lines are at divisible by two distances from each other.
• Two shapes are *adjacent* if l_∞ distance between them is equal to one and their projections on either a vertical or a horizontal axis intersect by a segment of length at least one.



Model E is not three colorable.

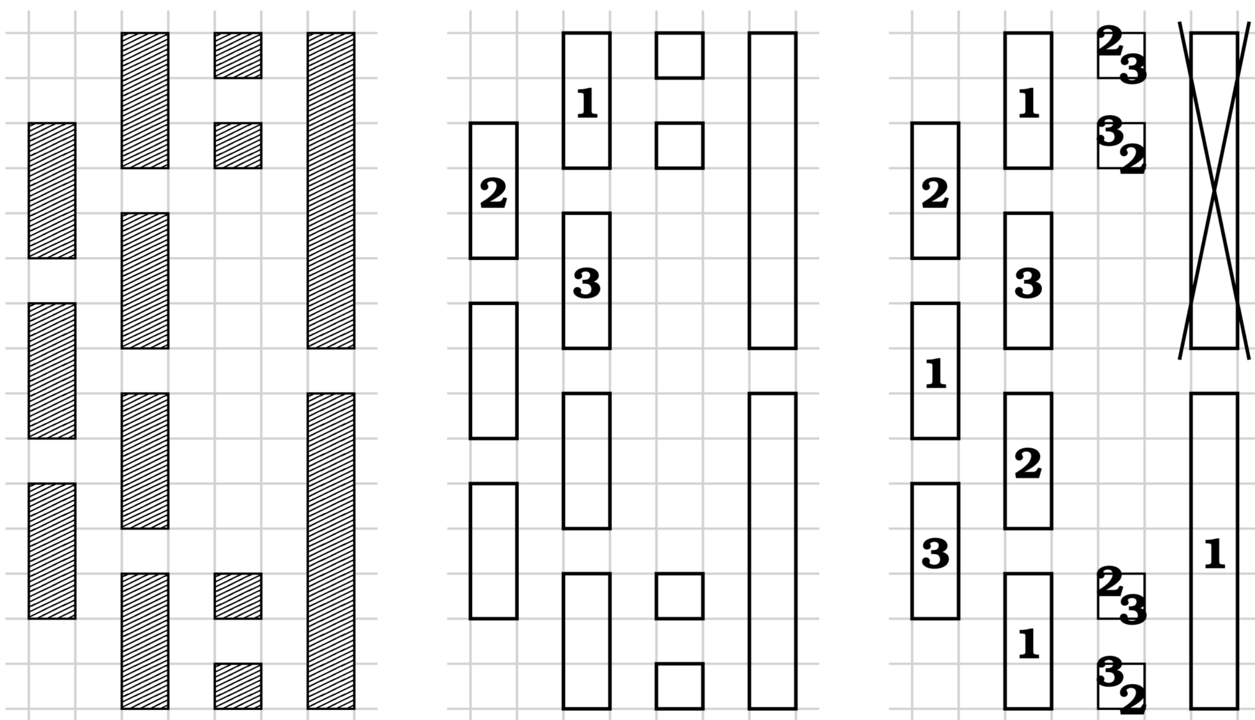


a b c

Model F. Add more constraints:

- A shape can have at most 2 shapes adj. to it from the left.
- A shape can have at most 2 shapes adj. to it from the right.

Still not three colorable:



a b c

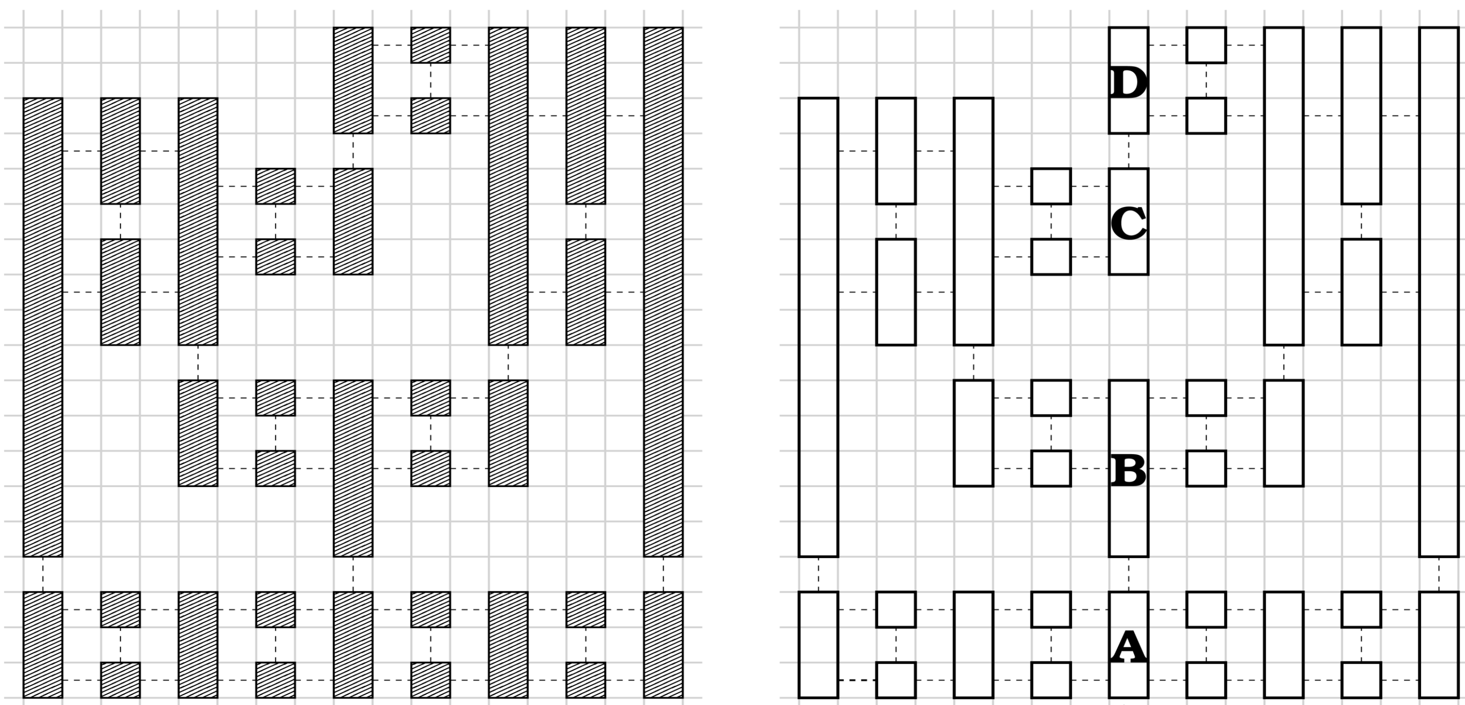
Model G: Add even more constraints.

- A shape can have at most 2 shapes adj. to it from the left.
- A shape can have at most 2 shapes adj. to it from the right.
- A shape can have at most 1 shape adjacent to it vertically.

Model G: Add even more constraints.

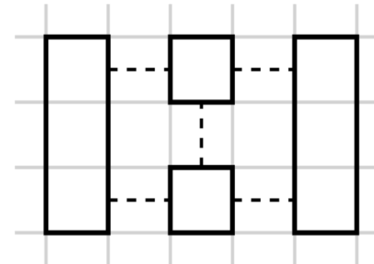
- A shape can have at most 2 shapes adj. to it from the left.
- A shape can have at most 2 shapes adj. to it from the right.
- A shape can have at most 1 shape adjacent to it vertically.

Now it is much harder to find a not colorable example.
But it exists. The model in still NOT colorable.

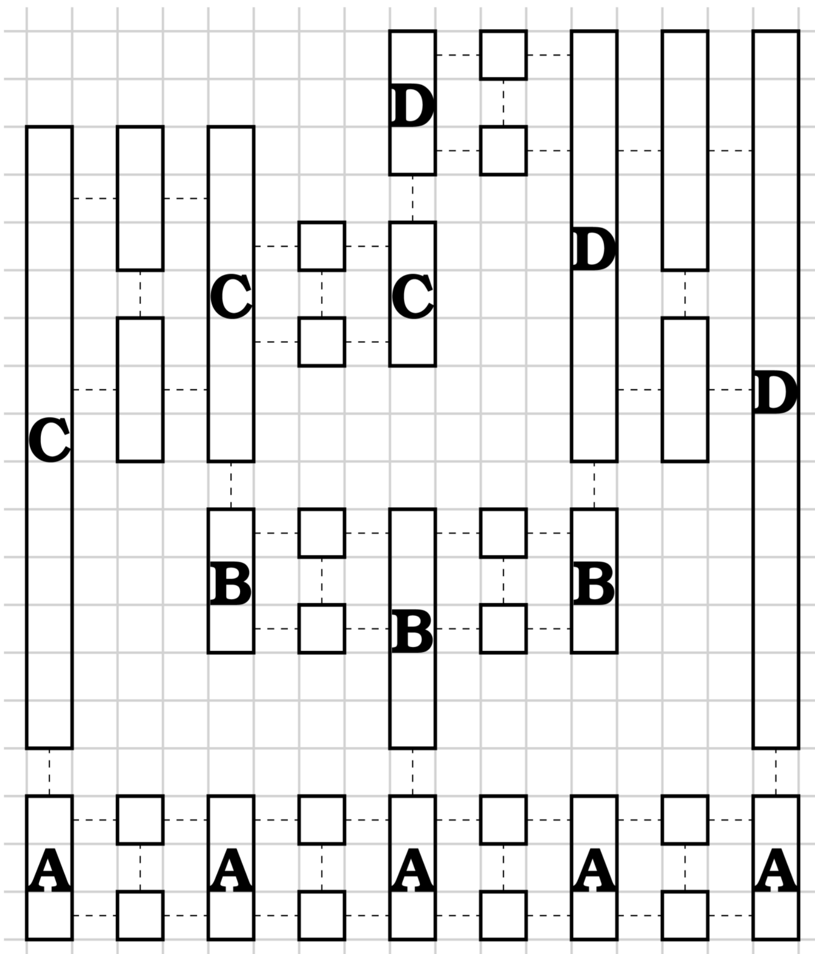


a b

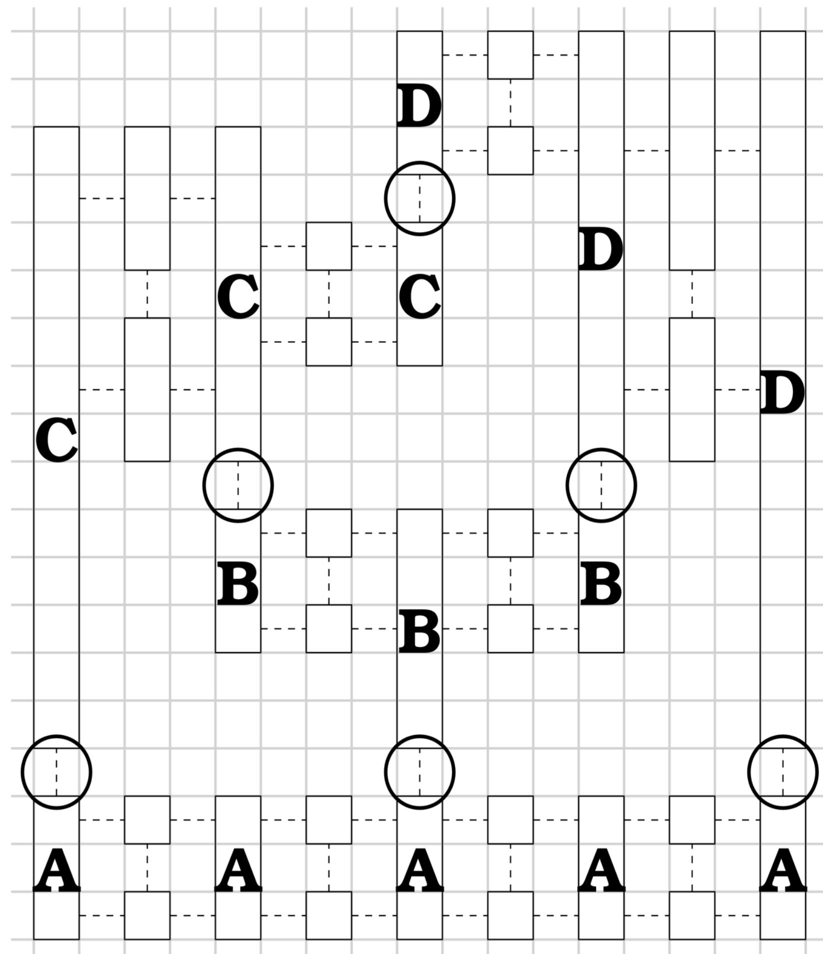
The leftmost and the rightmost shapes In this configuration must have one and the same color :



$$A \neq B, \quad A \neq C, \quad A \neq D, \quad B \neq C, \quad B \neq D, \quad C \neq D.$$



a



b

Any further reduction of limits on the number of horizontal or vertical interactions between shapes leads to a three-colorable model.

Model H:

- A shape can have at most 1 shape adj. to it from the left.
- A shape can have at most 2 shapes adj. to it from the right.
- A shape can have at most 1 shape adjacent to it vertically.

Conclusions and Future Research

- We analyzed triple and quadruple coloring of various layout models, with the goal of developing robust layout methodologies.
- Layout models that guarantees 3/4-colorability are presented
  - *O(n*log(n))* time complexity for coloring, making them suitable for practical layouts
  - Demonstrated that a slight relaxation can lead to un-colorability
- For correct-by-construction layout, we would like to explore further the correct-by-construction layout model
  - Analyzing further the complexity of triple and quadruple coloring of the graphs that result from layouts which belong to models that are not generally colorable but some individual layouts of which still can be colored.
  - Investigation on the standard-cell generation methodology using the generated models