Finding a Needle in the Haystack of Hardened Interconnect Patterns

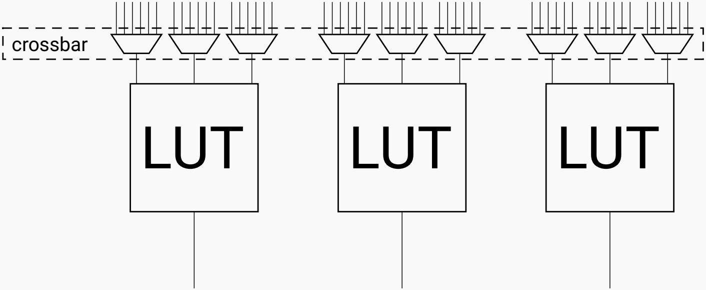
S. Nikolić, G. Zgheib*, and P. Ienne

FPL19, Barcelona, 09.09.2019

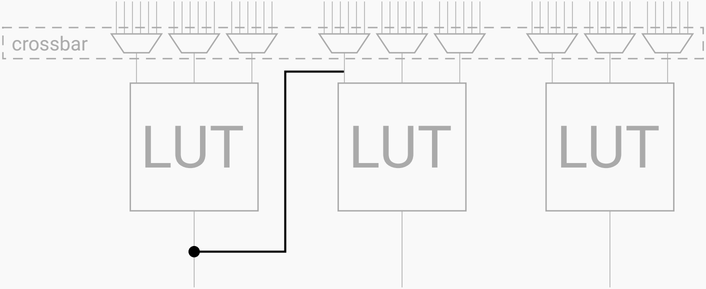
École Polytechnique Fédérale de Lausanne
*Intel Corporation

EPFL

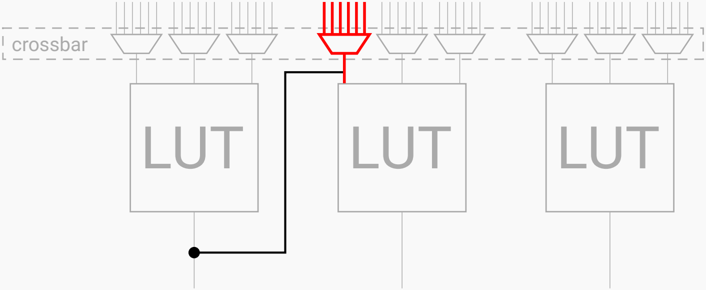
Why harden connections?



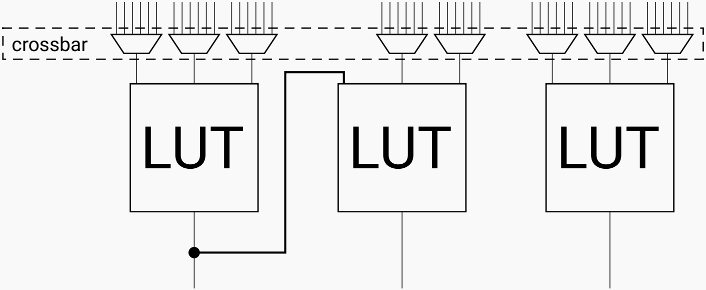
Why harden connections?



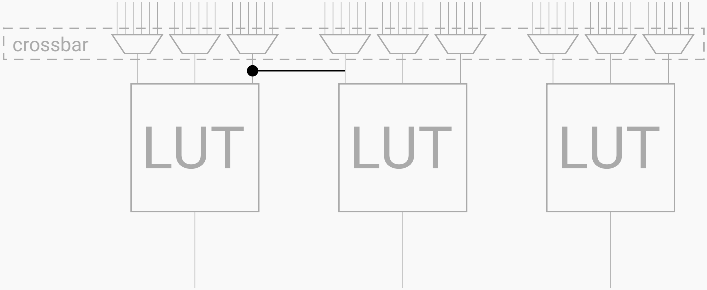
Why harden connections?



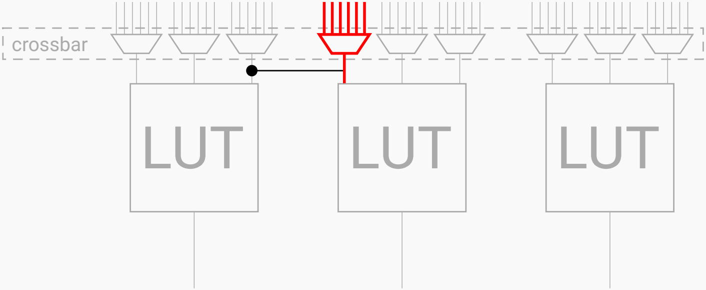
Why harden connections?



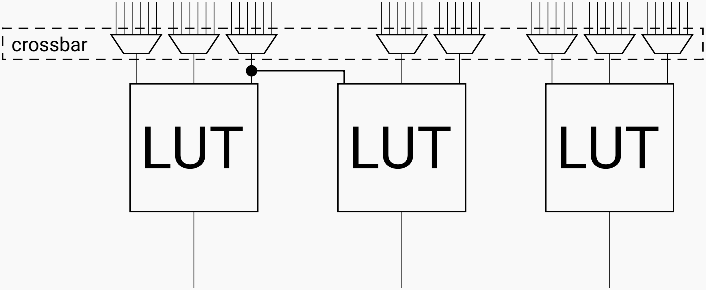
Why harden connections?



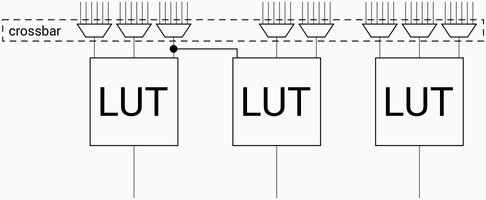
Why harden connections?



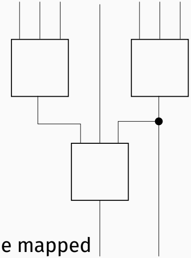
Why harden connections?



What is the price?

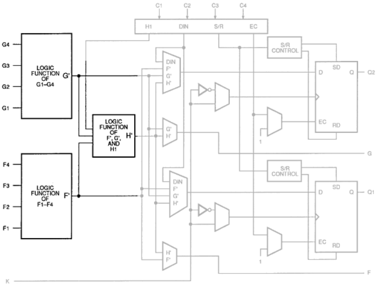


Cluster architecture

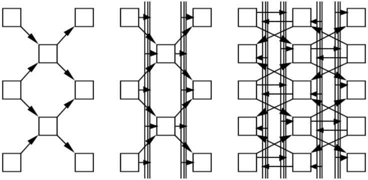


Circuit to be mapped

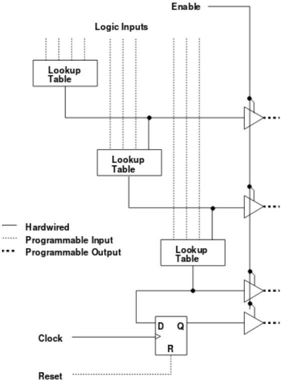
XC4000 [1]



Triptych [3]



UTFPGA1 [2]



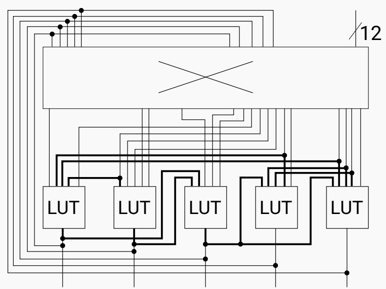
[1] H.-C. Hsieh, W. S. Carter, J. Ja, E. Cheung, S. Schreifels, C. Erickson, P. Freidin, L. Tinkey, and R. Kanazawa. Third-generation architecture boosts speed and density of field-programmable gate arrays, 1990

[2] P. Chow, S. O. Seo, D. Au, B. Fallah, C. Li, and J. Rose. A 1.2um CMOS FPGA using cascaded logic blocks and segmented routing, 1991

[3] C. Ebeling, G. Borriello, S. A. Hauck, D. Song, E. A. Walkup. TRIPTYCH: A New FPGA Architecture, 1991

Challenges

How to map on patterns?
(CAD tool scalability)

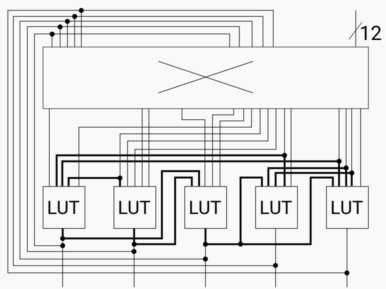


Challenges

How to design the patterns?

How to map on patterns?

(CAD tool scalability)

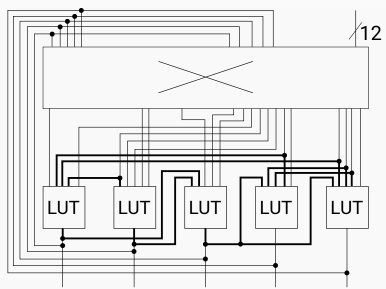


Challenges

How to design the patterns?

- Intuition?

How to map on patterns?
(CAD tool scalability)

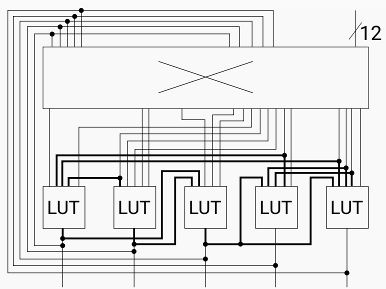


Challenges

How to design the patterns?

- Intuition?
- Enumeration

How to map on patterns?
(CAD tool scalability)



Challenges

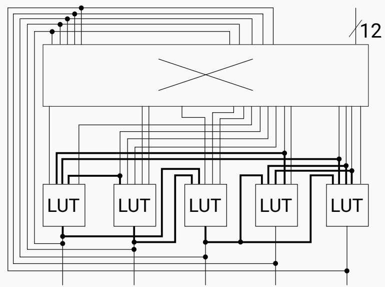
How to design the patterns?

- Intuition?
- Enumeration

$5 \times 5\text{-LUT} \sim 10^8$

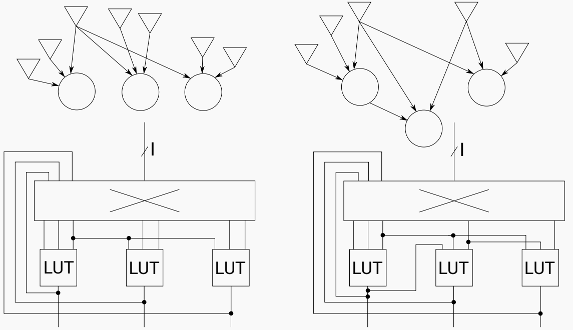
How to map on patterns?

(CAD tool scalability)

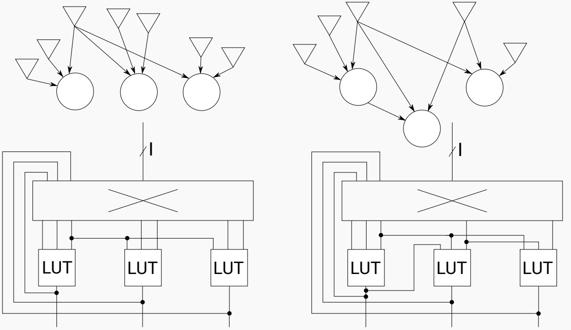


Enumeration

Representation

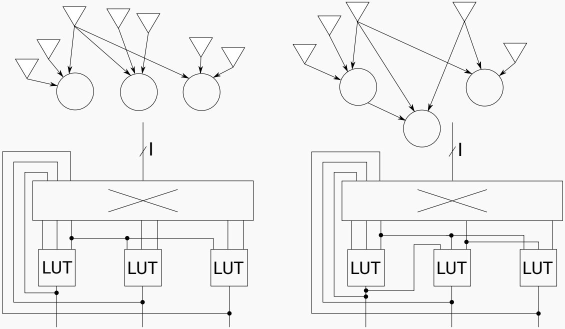


Representation



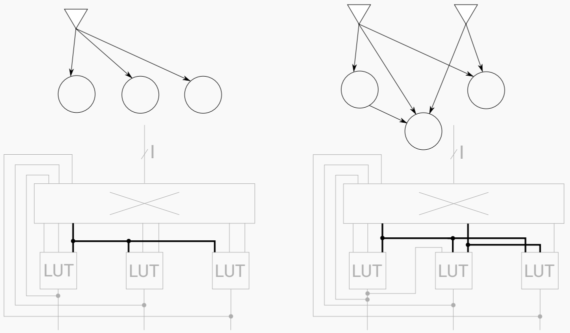
- represent each LUT by a node (circles)

Representation



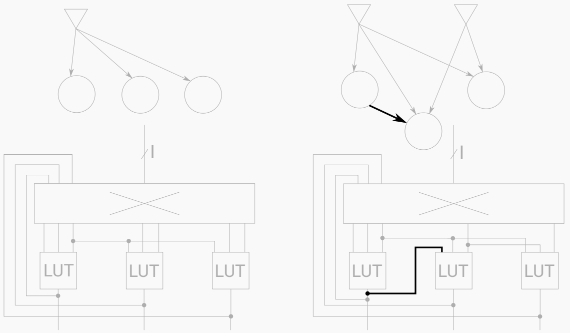
- represent each LUT by a node (circles)
- only represent shared inputs (triangles)

Representation



- represent each LUT by a node (circles)
- only represent shared inputs (triangles)

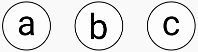
Representation



- represent each LUT by a node (circles)
- only represent shared inputs (triangles)
- each edge is a hardened connection

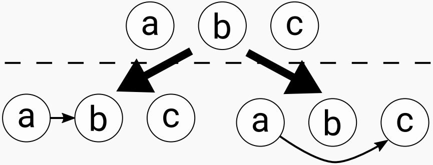
Enumeration (no input sharing for now)

```
//V - vertex set

G = (V, {})
expandable = (G)
while expandable {
  G = pop(expandable)
  for e in V x V {
    if keep(G + e) {
      push(G + e, expandable)
    }
  }
}
```



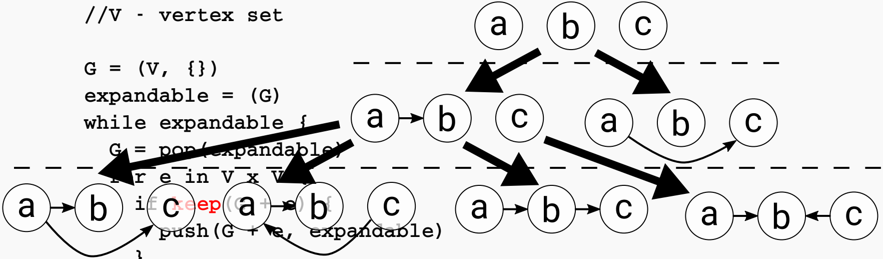
Enumeration (no input sharing for now)

```
//V - vertex set

G = (V, {})
expandable = (G)
while expandable {
  G = pop(expandable)
  for e in V x V {
    if keep(G + e) {
      push(G + e, expandable)
    }
  }
}
```



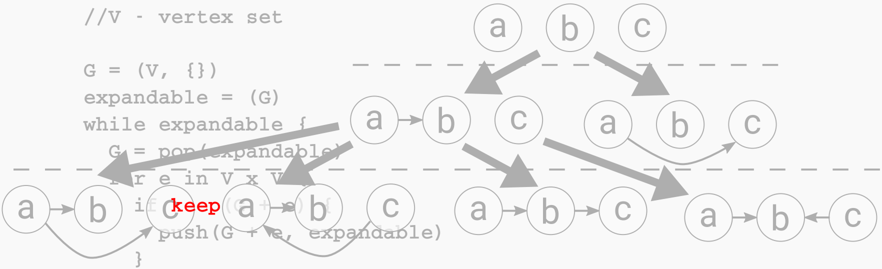
Enumeration (no input sharing for now)

```
//V - vertex set

G = (V, {})
expandable = (G)
while expandable {
  G = pop(expandable)
  for e in V x V {
    if keep(G + e) {
      push(G + e, expandable)
    }
  }
}
```



Enumeration (no input sharing for now)

```
//V - vertex set

G = (V, {})
expandable = (G)
while expandable {
  G = pop(expandable)
  for e in V x V {
    if keep(G + e) {
      push(G + e, expandable)
    }
  }
}
```



When to stop?

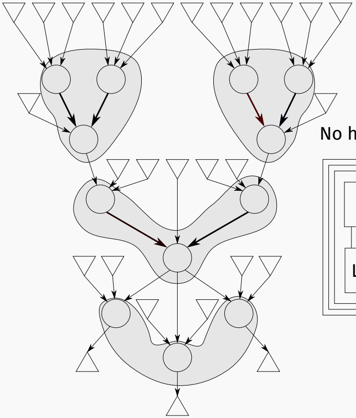
When area or delay stop decreasing?

When to stop?

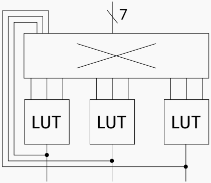
When area or delay stop decreasing?
When area or delay start increasing?

When to stop?

Circuit to be mapped

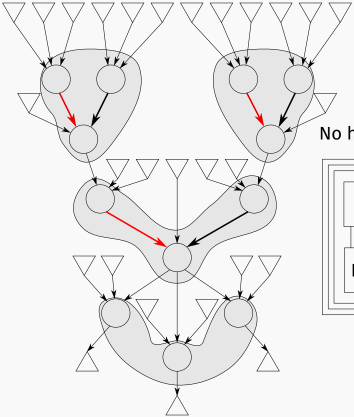


No hardened connections

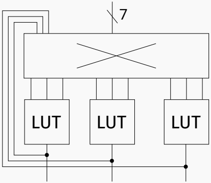


When to stop?

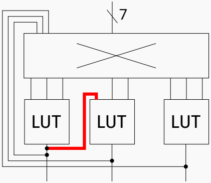
Circuit to be mapped



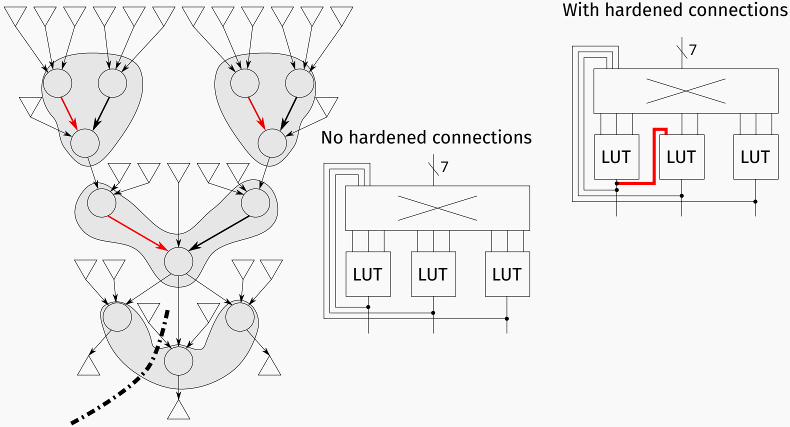
No hardened connections



With hardened connections



When to stop?

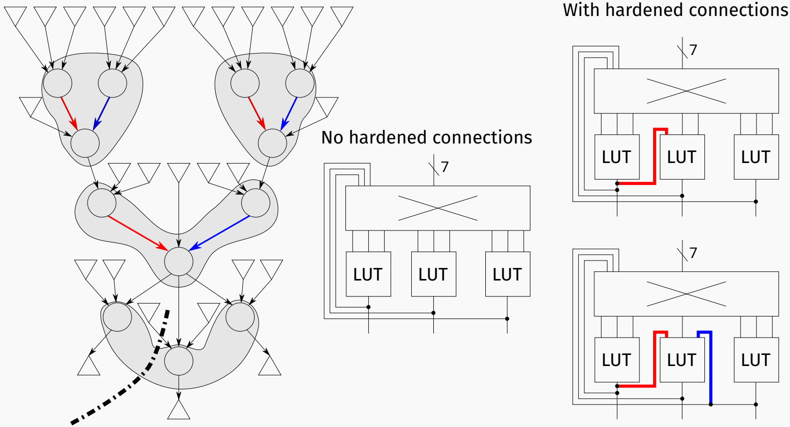


Circuit to be mapped

No hardened connections

With hardened connections

When to stop?



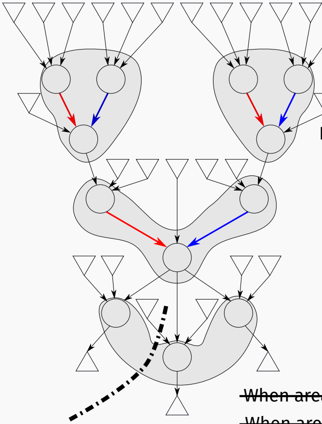
Circuit to be mapped

No hardened connections

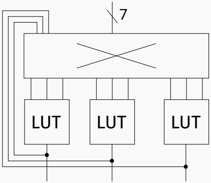
With hardened connections

When to stop?

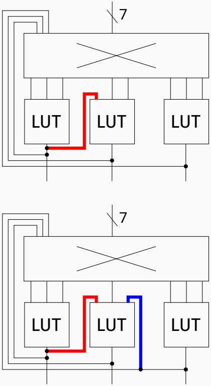
Circuit to be mapped



No hardened connections



With hardened connections

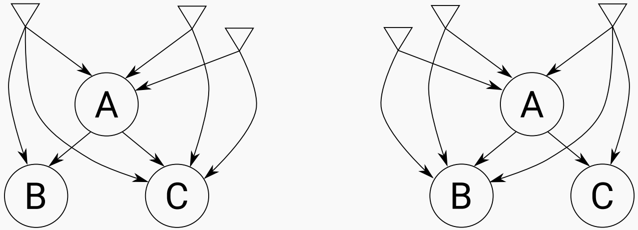


~~When area or delay stop decreasing?~~

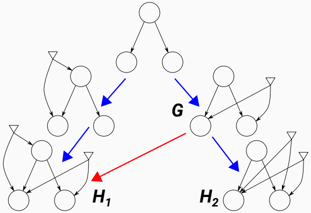
~~When area or delay start increasing?~~

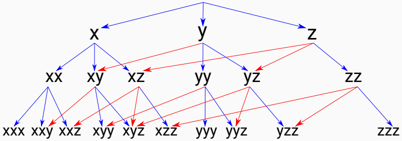
When to stop?

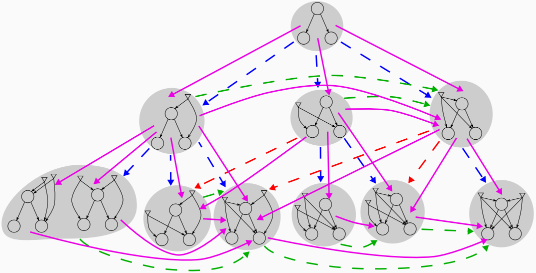
Other issues: avoiding listing duplicates



Other issues: maintaining subgraph relations







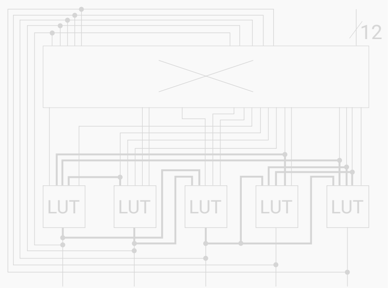
Challenges

How to design the patterns?

- Intuition?
- Enumeration

  $5 \times 5\text{-LUT} \sim 10^8$

How to map on patterns?
(CAD tool scalability)



Experiments

Setup

Setup

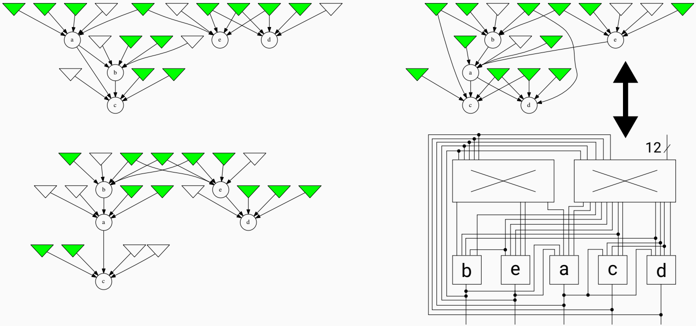
- Search space: acyclic five 5-LUT patterns ($\sim 10^8$ patterns)

Setup

- Search space: acyclic five 5-LUT patterns ($\sim 10^8$ patterns)
- Architecture = 4x the pattern with a shared crossbar (20 5-LUT clusters)

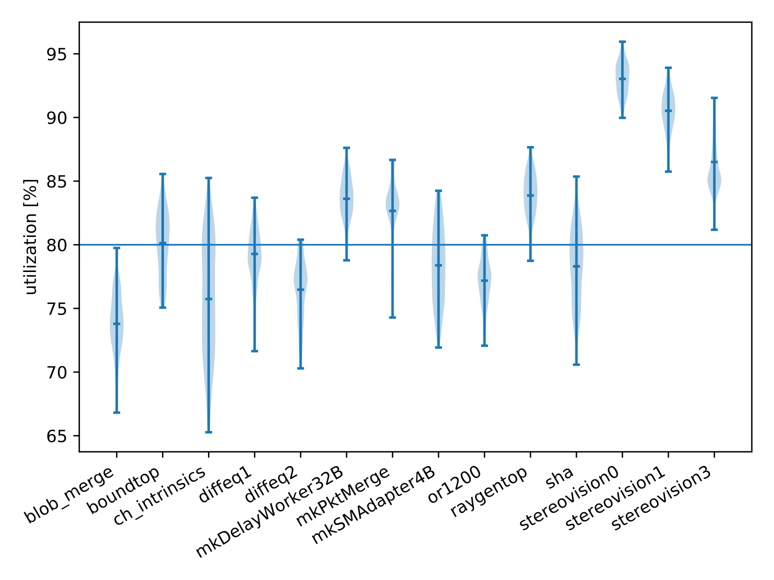
Results

Some examples



Found 261 patterns with only 12 external inputs achieving $\sim$ 80% packing density

Results



utilization [%]
95
90
85
80
75
70
65
blob_merge
boundtop
ch_intrinsics
diffeq1
diffeq2
mkDelayWorker32B
mkPktMerge
mkSMAdapter4B
or1200
raygentop
sha
stereovision0
stereovision1
stereovision3

Conclusions

Numerical results not satisfactory (18-29% critical path delay increase)

But...

We have an efficient way of searching for good patterns

- searched the space $\sim 10^8$ in $<$ 12h
- search techniques completely independent of the mapping algorithms

In the future, this should help us understand what makes a good pattern and profit from connection hardening to the fullest

Thank you for attention

For questions, please see the poster