



Shashwat Shrivastava, Stefan Nikolić, Sun Tanaka, Chirag Ravishankar[‡], Dinesh Gaitonde[‡], and Mirjana Stojilović

EPFL

*University of Novi Sad

‡AMD

Routing Failures

Many designs take long and still fail to route

AMD VIVADO
ERROR: [Route 35-3] Design is **not routable** as its global congestion level is 6
Time (s): cpu = 10:18:07; elapsed = 10:19:43. Memory (MB): peak = 15725.180; gain = 2896.867; free physical = 793741; free virtual = 1259249

Is it due to the routing algorithm or poor routability of the FPGA architecture?

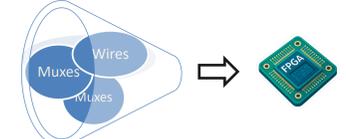
Typical Convergence Fix

FPGA user level



Tuning parameters of the routing algorithm

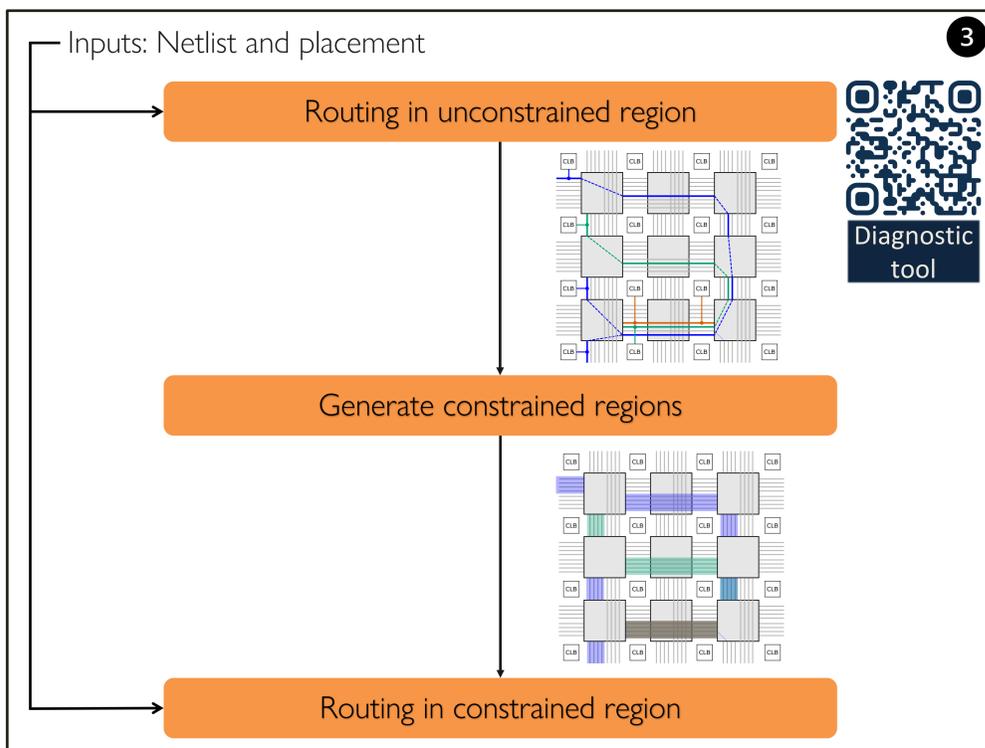
FPGA vendor level



Adding routing resources or modifying the architecture

But little focus on finding fundamental inefficiencies in the routing algorithm

Diagnostic Methodology



Routing in Constrained Region

Expectation:

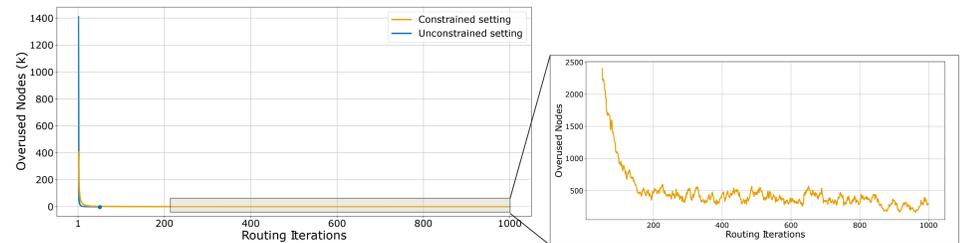
Reduced search space → route quickly compared to unconstrained setting

Path is guaranteed → should find the paths

Reality:

Does not even converge to a routing solution

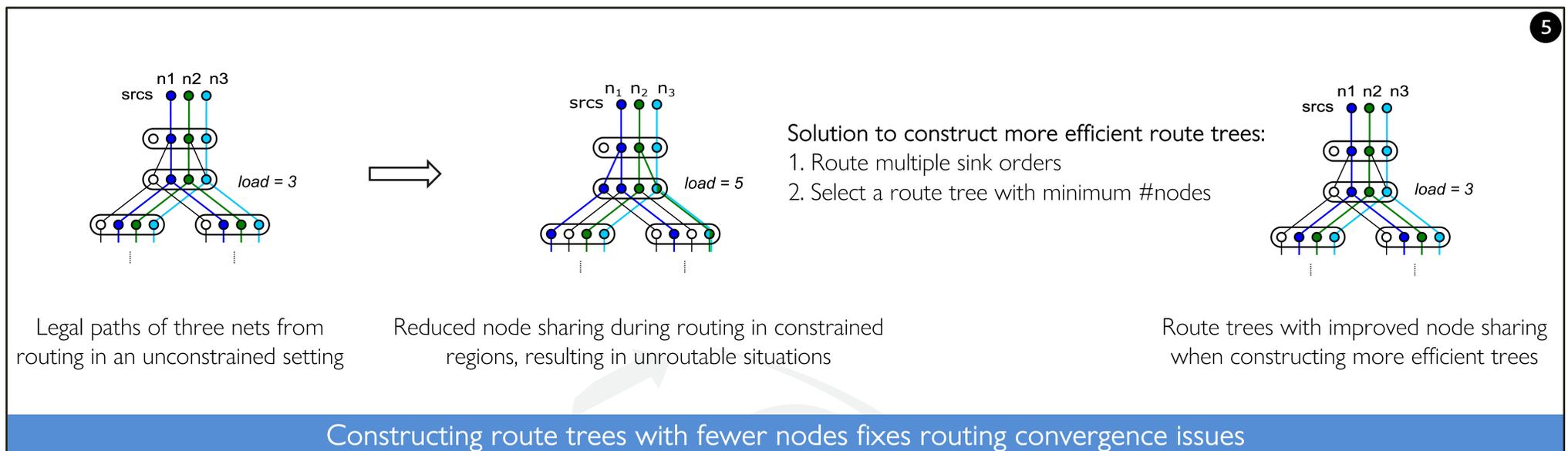
Example of routing one of the biggest benchmarks from ISPD16 suite



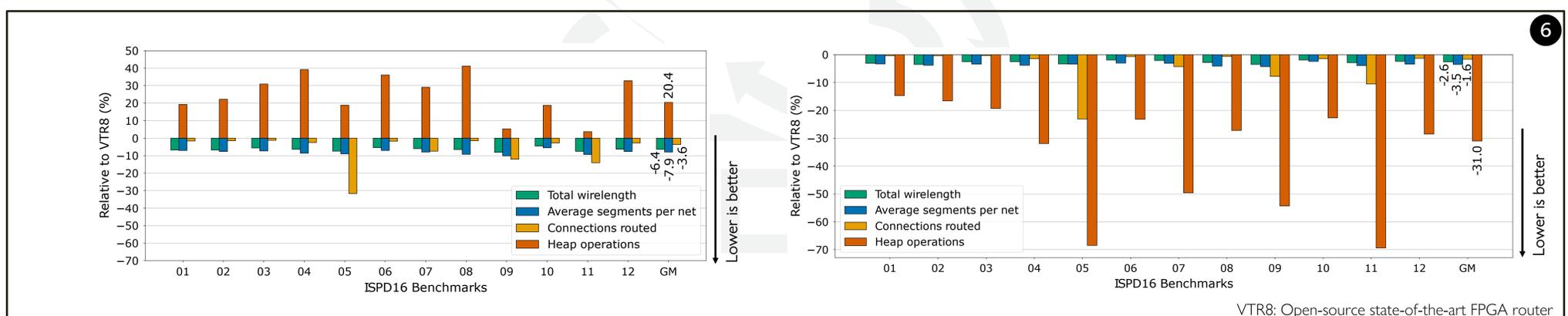
A routing convergence paradox!

Even with a guaranteed solution, benchmarks fail to converge

Where is the Problem?



Benefits of Constructing More Efficient Route Trees in the Unconstrained Setting



Constructing more efficient route trees holds the potential to improve routing runtime and quality of results