# IIBLAST: Speeding Up Commercial FPGA Routing by Decoupling and Mitigating the Intra-CLB Bottleneck

**Shashwat Shrivastava, Stefan Nikolić, Chirag Ravishankar, Dinesh Gaitonde, and Mirjana Stojilović**

**EPFL**

**Impact of Commercial IIB**

Routing through a commercial IIB closely resembling an AMD UltraScale FPGA (IIB-6-2L, two layers of 6:1 mux) vs a fully connected IIB (Full IIB)

## Routing vs Placement Runtime

<table>
<thead>
<tr>
<th>Threads</th>
<th>Placement</th>
<th>Routing</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40.6%</td>
<td>55.6%</td>
</tr>
<tr>
<td>8</td>
<td>36.1%</td>
<td>53.5%</td>
</tr>
</tbody>
</table>

Routing is the major compilation bottleneck

## Proposed Approach: Multi-Stage Router

**Step 1:** Routing until the IIB boundaries

**Step 2:** Partial routing solutions, IIB routing problems

**Step 3:** Combining partial solutions

- **Legal and complete?** Yes
- **No**
  - Incremental routing
  - Final routing solution

## Multi-Stage Router vs Single-Stage Router

Geomean runtime improvement 1.5×

**What is increasing Step 3 Runtime?**

Only 66% of IIBs legalize in Step 2, on average

## Low legalization rate increases runtime of step 3

Techniques to increase legalization rate:
- Route a net driving a FF through a free LUT \( \rightarrow \) LUT-1
- Increase IIB mux size from 6:1 to 8:1 \( \rightarrow \) IIB-8-2L

Geomean runtime improvement 5×

**Recovered the runtime improvement from 1.5× to 5×**

---

**Impact of Commercial IIB**

Routing through commercial IIB is 6× slower

**Multi-Stage Router**

- Step 3 consumes significant fraction of runtime

**Geomean runtime improvement 1.5×**

**Step 3** consumes significant fraction of runtime

What is increasing Step 3 Runtime?

Only 66% of IIBs legalize in Step 2, on average

**Low legalization rate increases runtime of step 3**

Techniques to increase legalization rate:
- Route a net driving a FF through a free LUT \( \rightarrow \) LUT-1
- Increase IIB mux size from 6:1 to 8:1 \( \rightarrow \) IIB-8-2L

Geomean runtime improvement 5×

Recovered the runtime improvement from 1.5× to 5×