Looking at the bigger picture

The Problem with Combinatorial Approaches: The Search Effort

- Combinatorial approaches are often slower due to the large search effort
- For a constraint-based compiler, that would imply long compilation times

Base Constraints for Instruction Scheduling

- Precedence Constraint
  - An instruction $x$ may not start execution before its predecessor $p$ was issued and its latency has passed:
    $$c_x \geq c_p + \text{lat}(p)$$
  - with $c_x$ denoting the issue cycle
  - $\text{lat}(\cdot)$ denoting the latency
  - The precedence constraint holds for the example on the left.

Applying the Constraint

- Figure shows encountered failed nodes for base and extended model during solution search for a basic block
- Extended model with predecessor constraints significantly cuts down the number of failed nodes (in total for 291 basic blocks)
- Achieves to find optimal solutions for four more basic blocks within a time limit of 30 seconds

References