

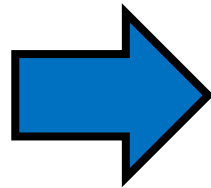
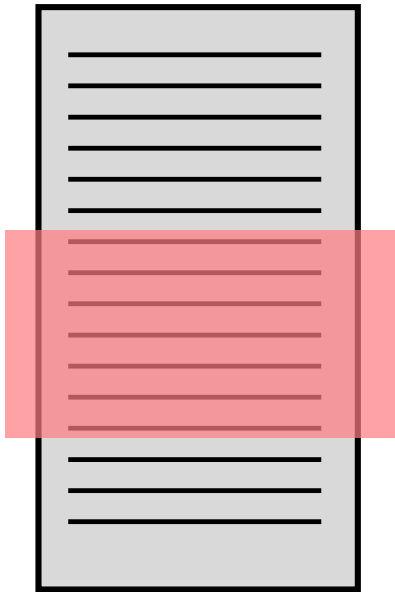
# CHAINSAW

## Von-Neumann Accelerators To Leverage Fused Instruction Chains

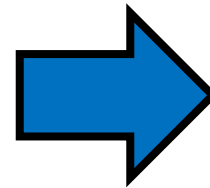
Amirali Sharifian, **Snehasish Kumar**, Apala Guha,  
Arrvindh Shriraman

# AXC Challenge 1: Idleness

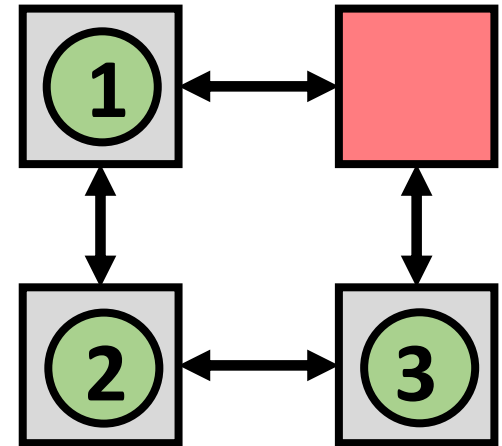
**Application**



**DFG**



**Spatial Fabric**

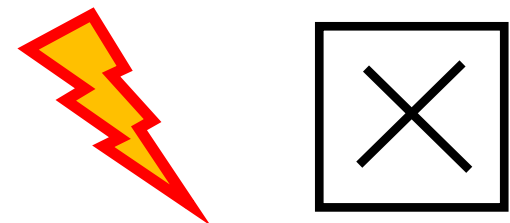
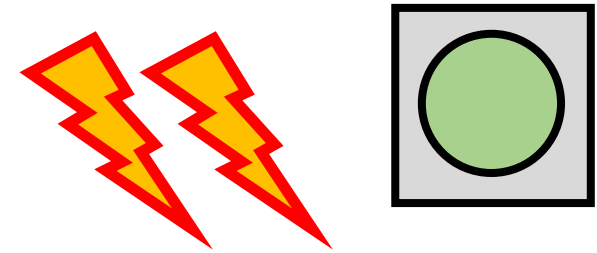
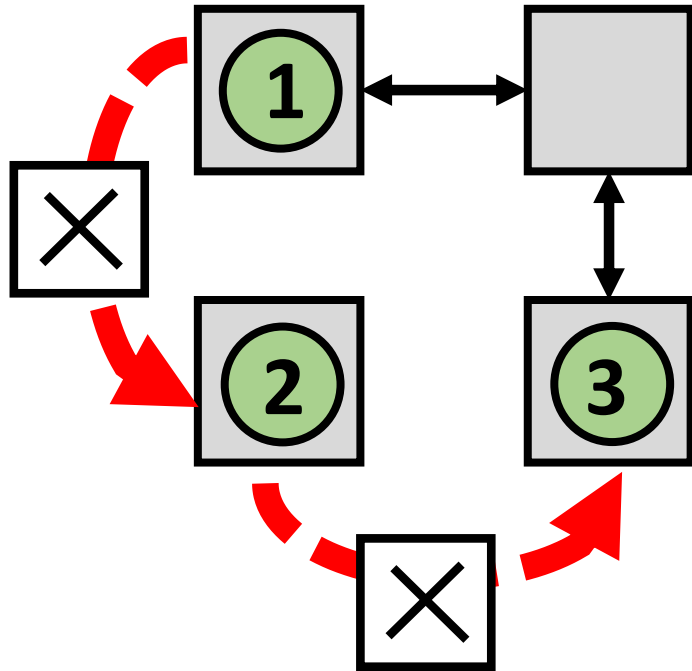


 Fabric Size  $\propto$  Dataflow graph size

Larger dataflow (possibly more idleness)

# AXC Challenge 2: Data movement

## Spatial Fabric



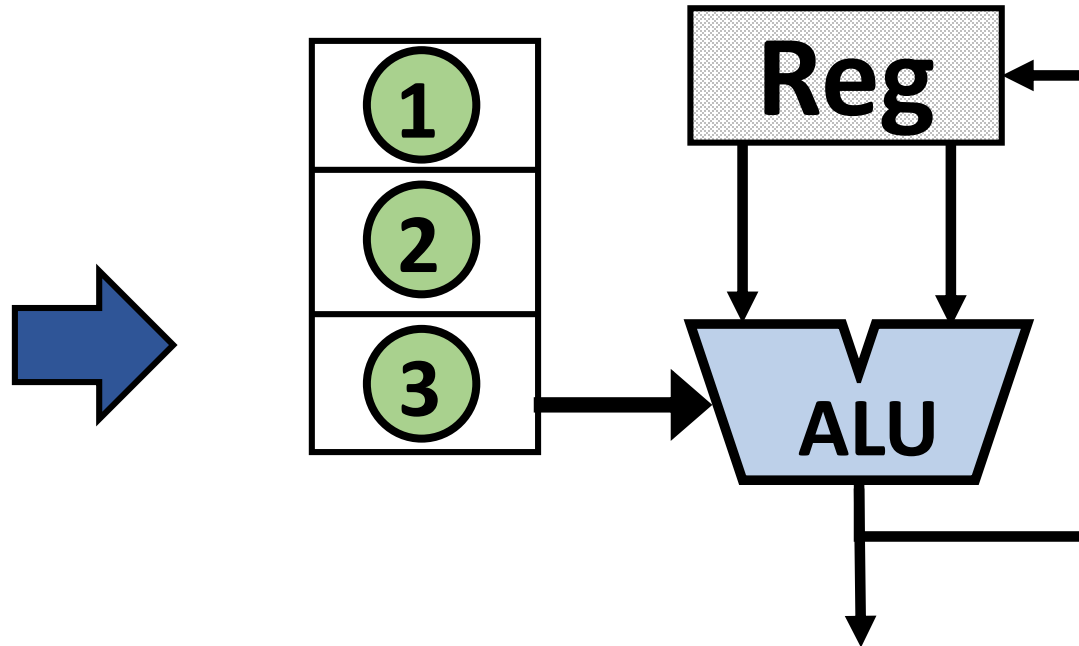
50% Energy overhead for data movement

# Von-Neumann Features

**DFG**



**Ins. Buffer**



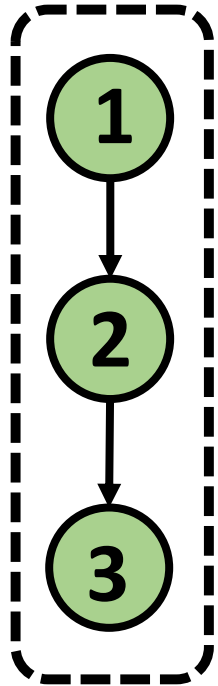
Temporal Mapping = Less Idleness

Central Register File

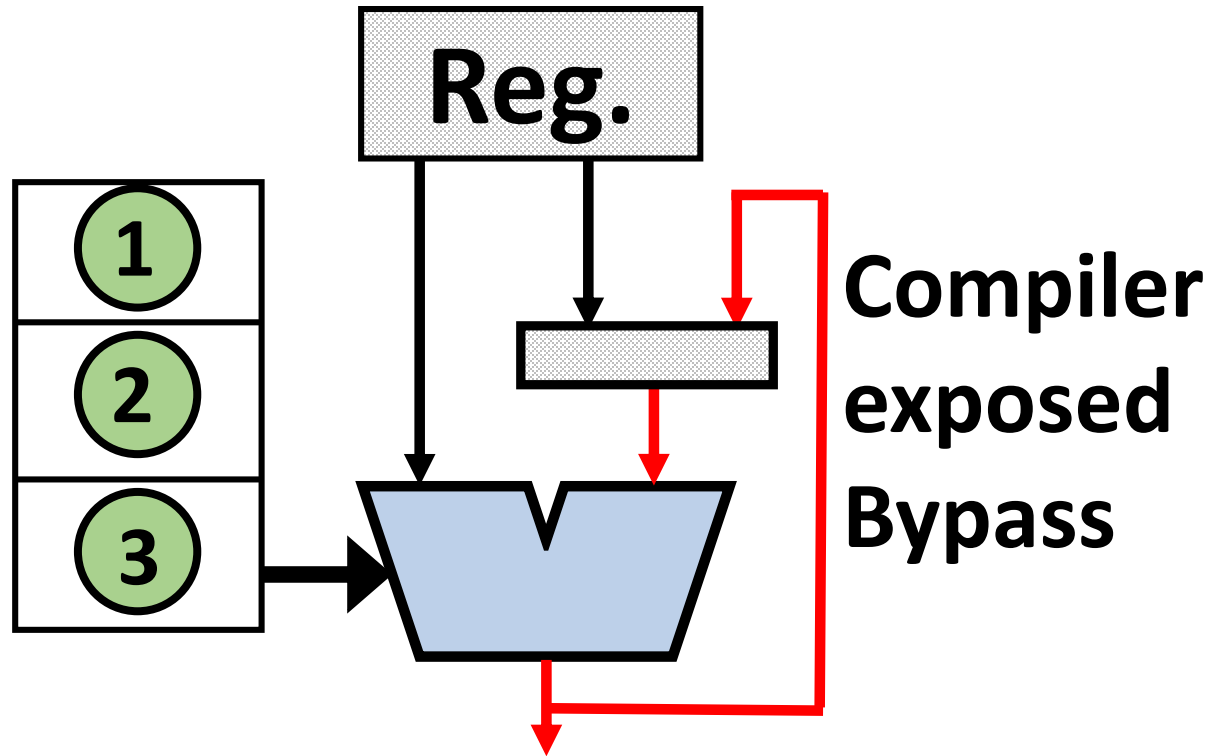
Fetch and Decode

# Our Approach : Fused Instruction Chains

## CHAIN DFG



## Von-Neumann + Chains



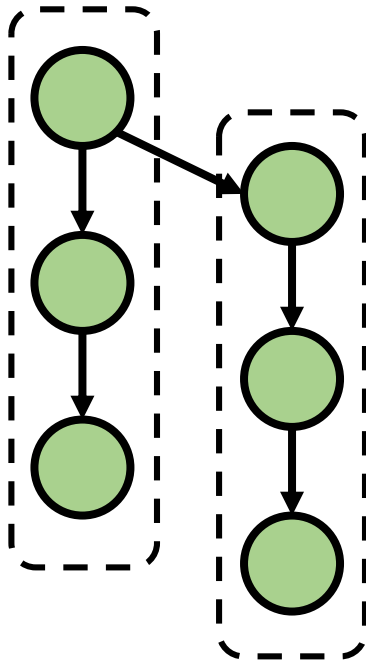
Temporal Mapping = Less Idleness  
Bypass = Internalize communication

# Our Approach : Fused Instruction Chains

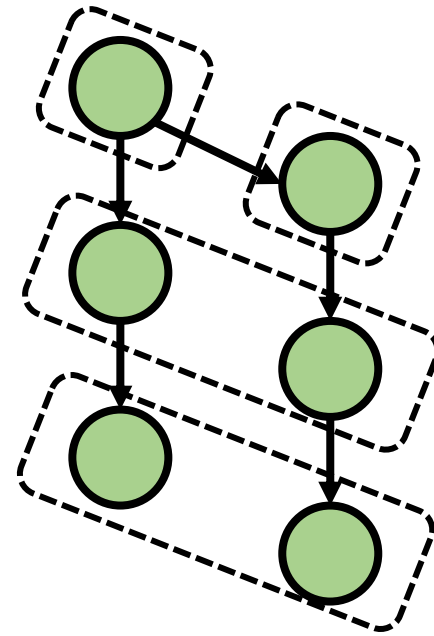
- Do chains exist in a DFG?
- How to form the chains?
- What are the challenges?
- Modeling and Evaluation

# CHAINS vs VLIW

## CHAINSAW



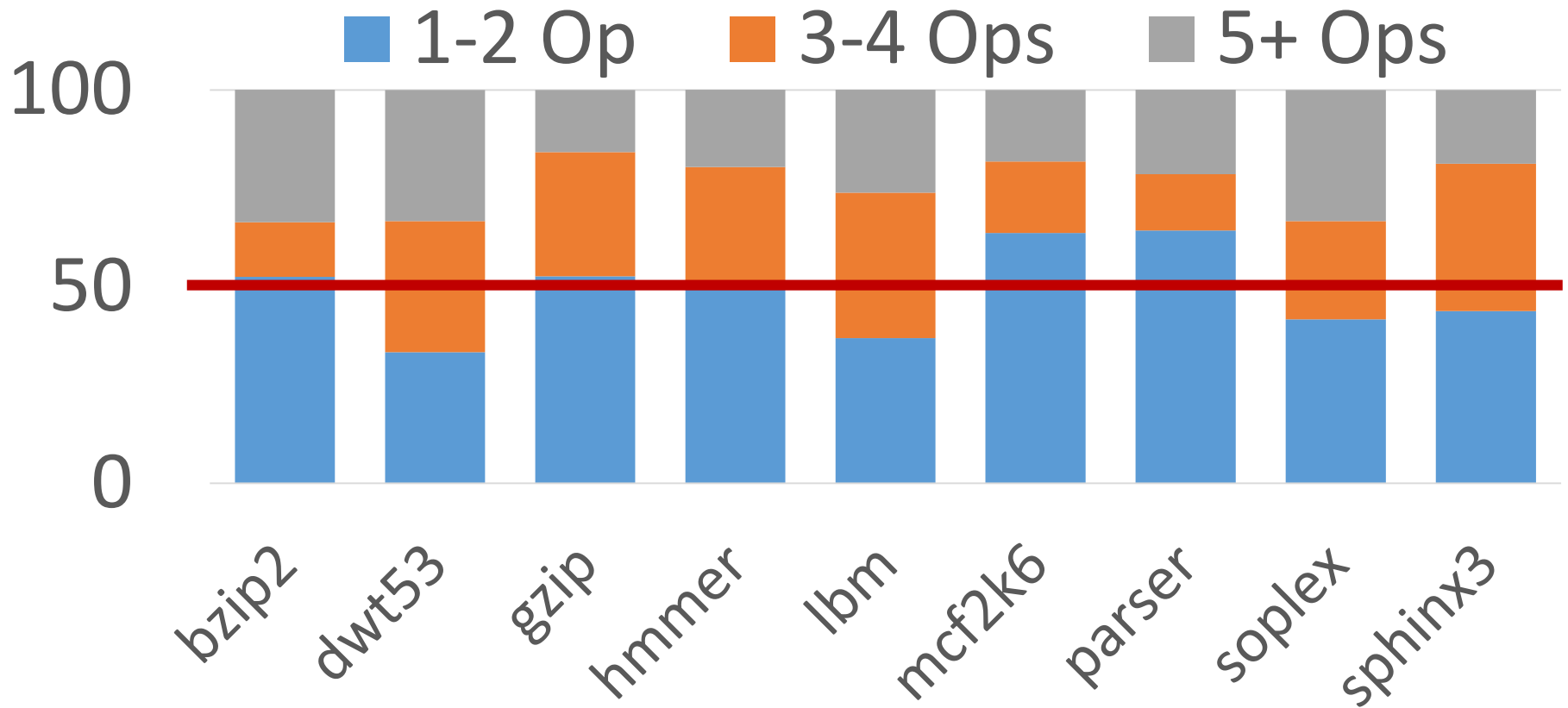
## VLIW



Finding dependent  
instructions  
Vertical Fusion

Finding independent  
instructions  
Horizontal Fusion

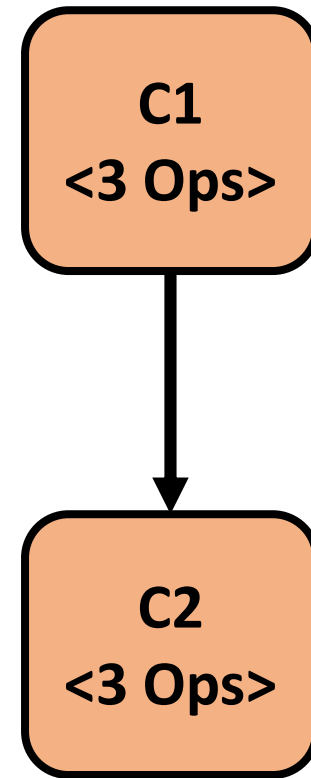
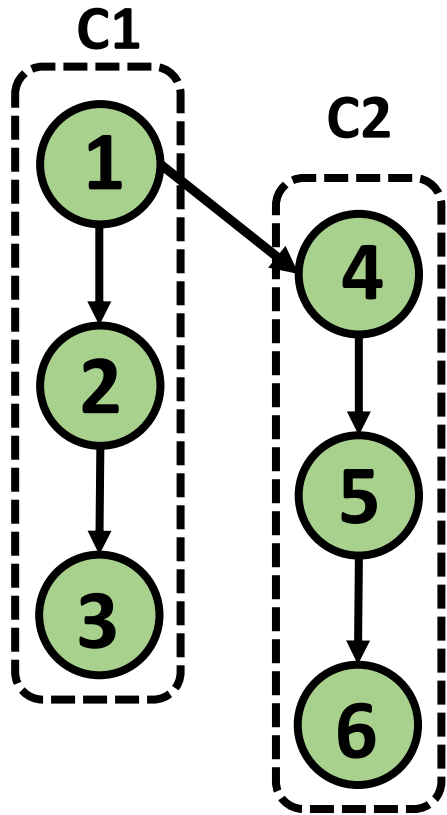
# Do chains exist in a DFG ?



50–80% of DFG part of 3+ op chains

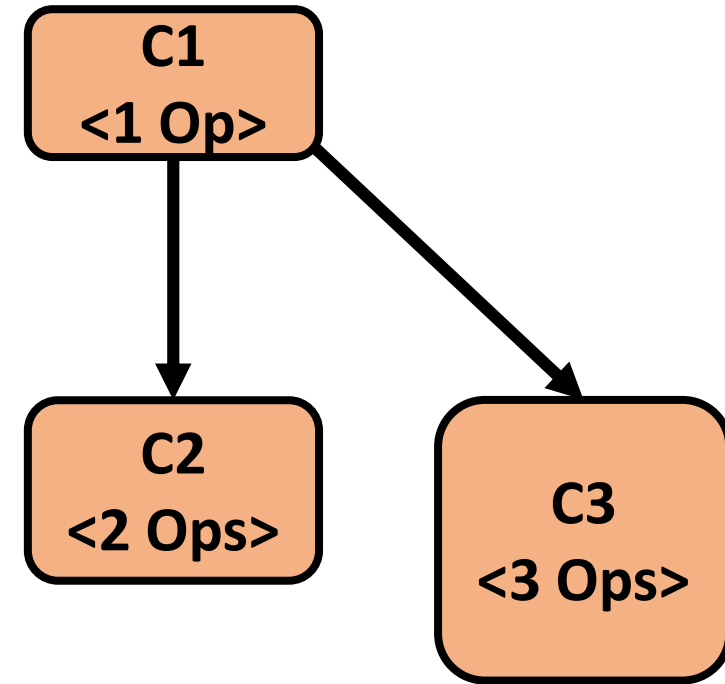
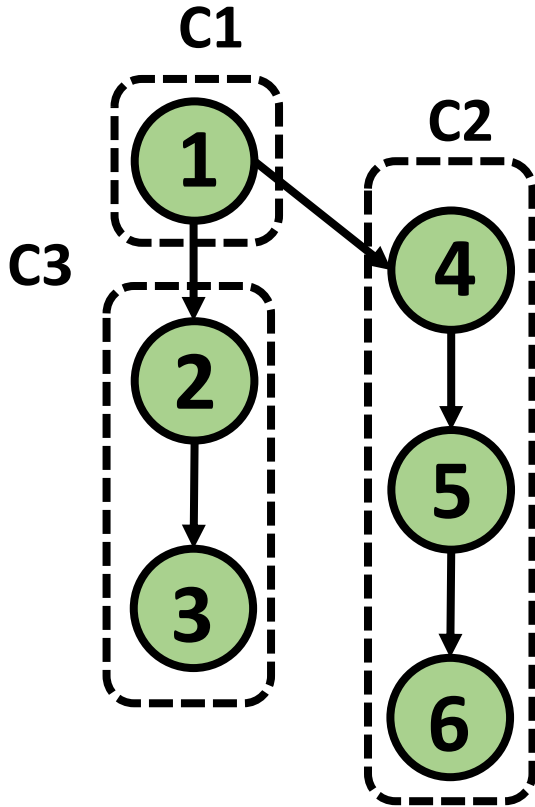


# How to form chains? Reduce Comm.



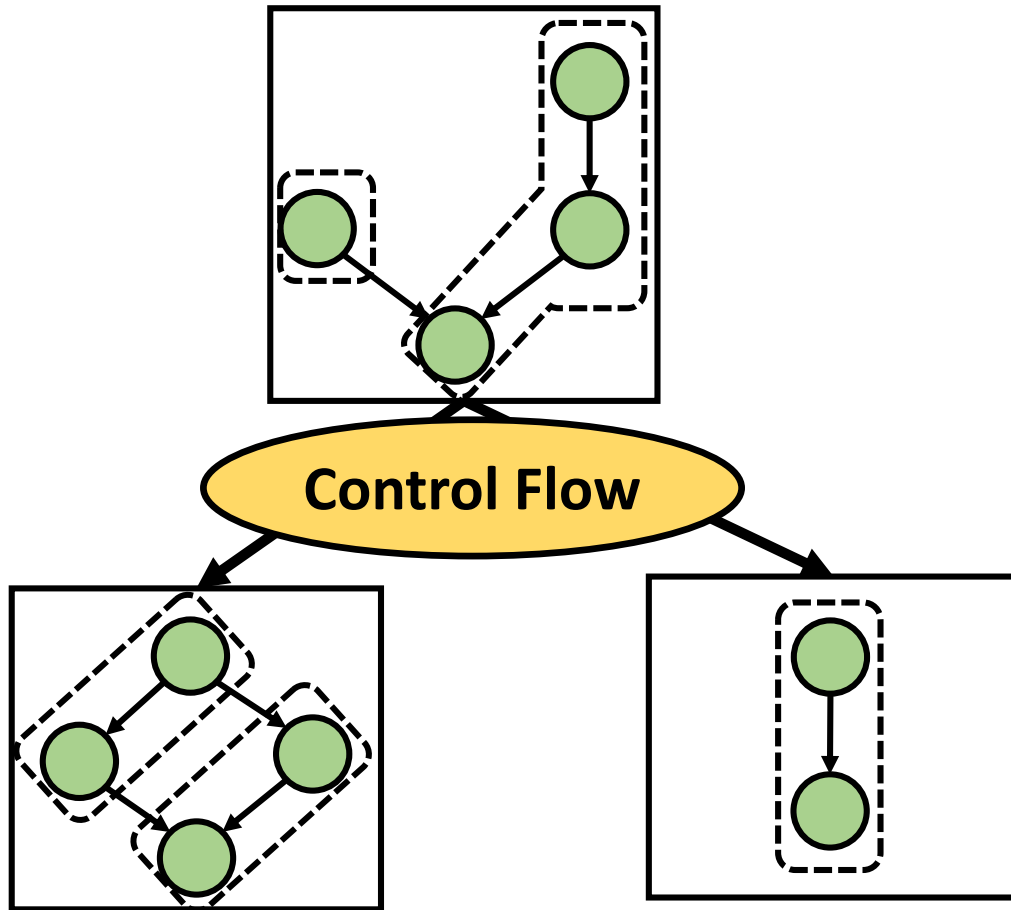
Internalizing communication  
Reducing Program ILP

# How to form chains? Optimize for ILP

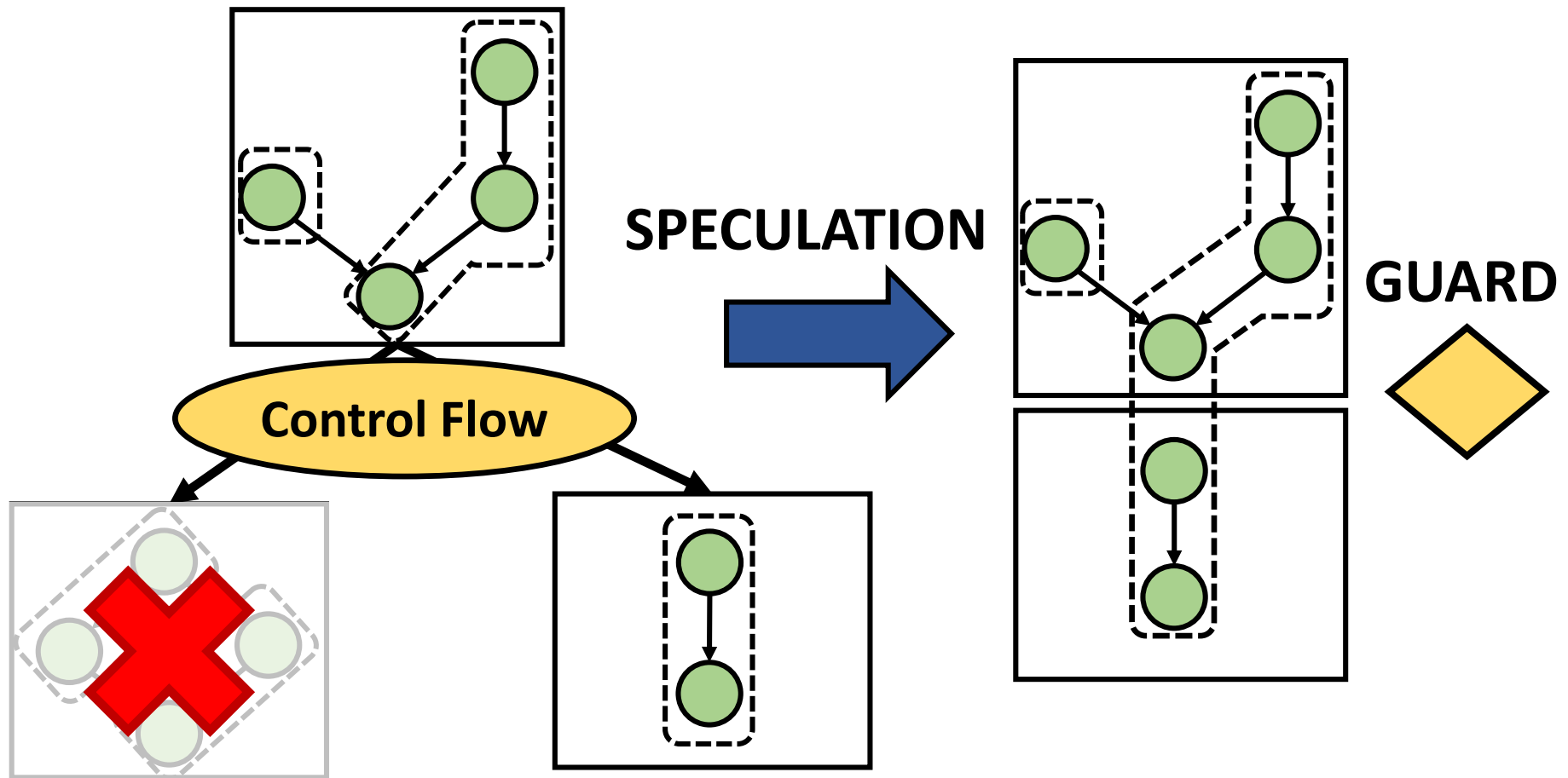


Having same ILP as the prog.  
Increasing communication

# How to extract – *longer* – Chains



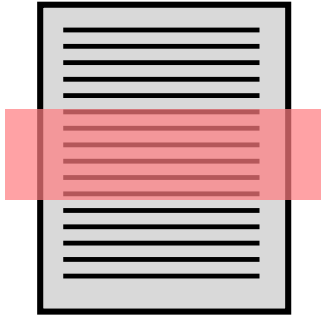
# How to extract – *longer* – Chains



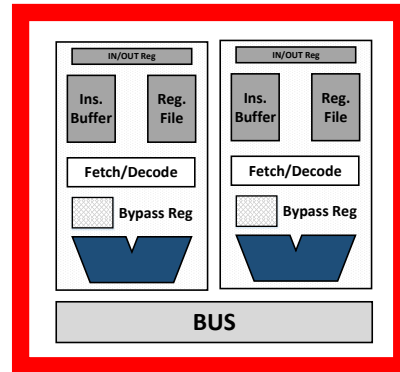
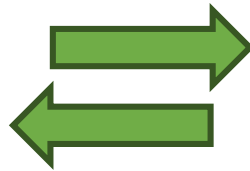
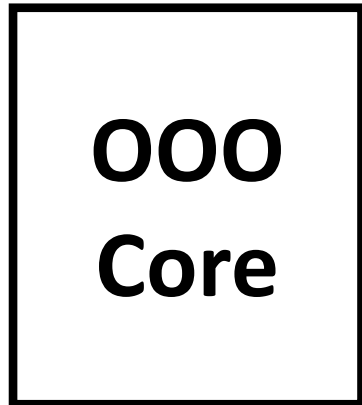
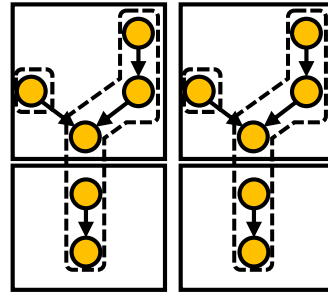
Larger Superblocks/Paths  $\Rightarrow$  Larger chains

# CHAINSAW is an Accelerator

## WORKLOAD



## HOT PATH

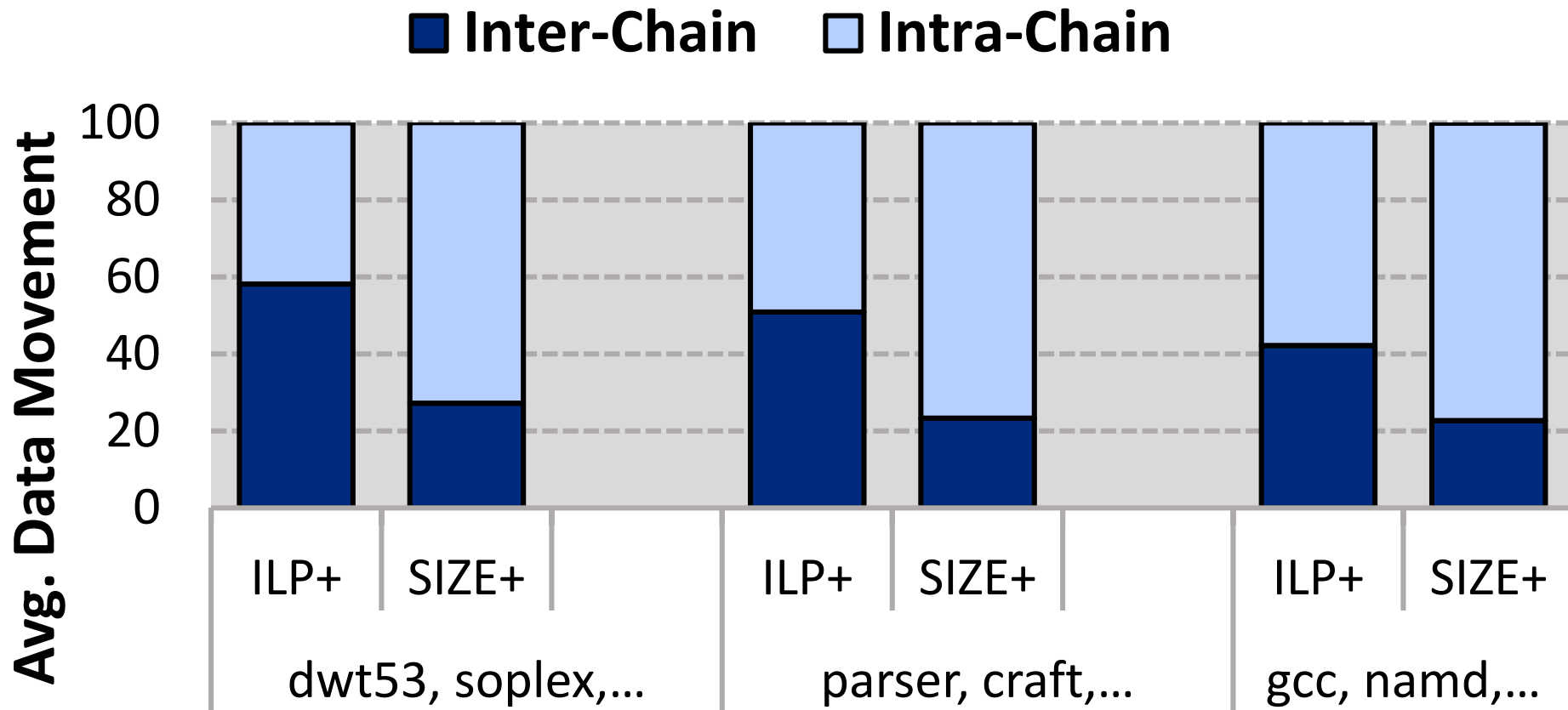


**CHAINSAW**

- Control free
- Only hot paths
- Limited inst. buffer

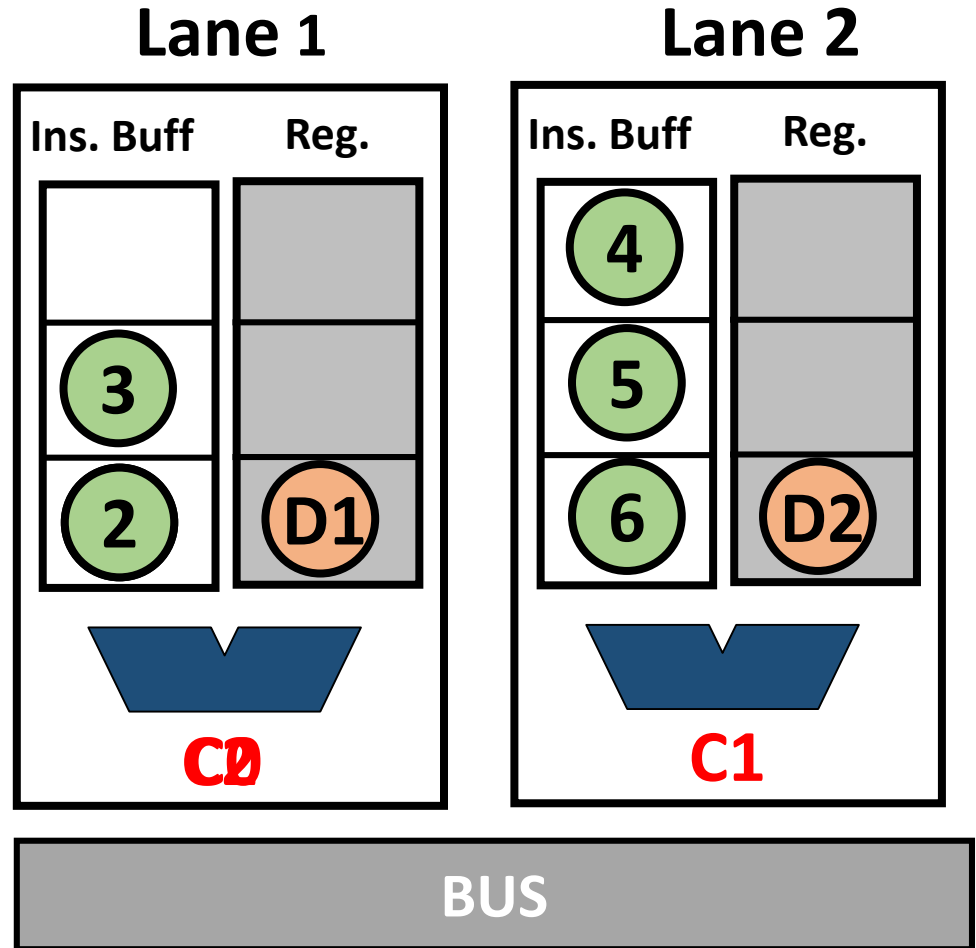
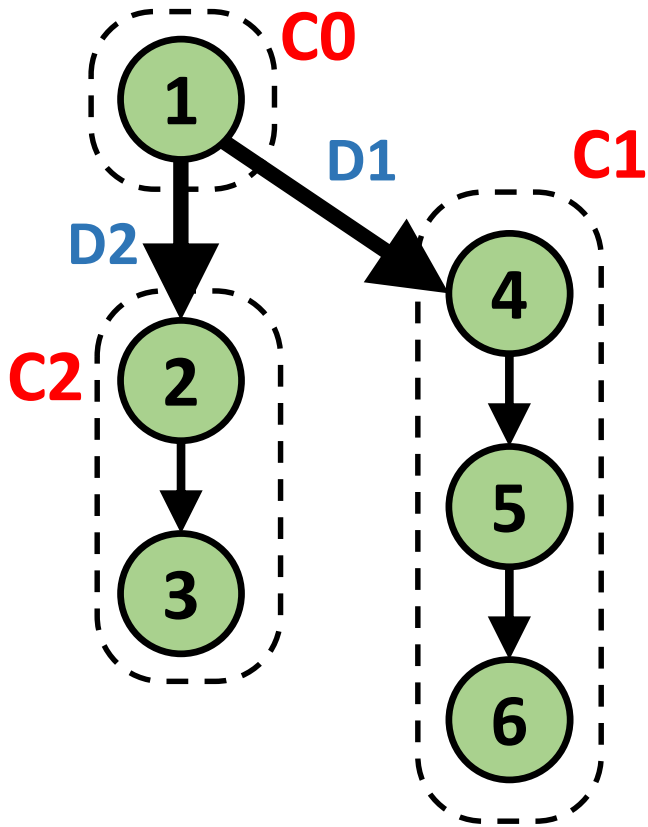
Cache Mem.

# How much can we localize?

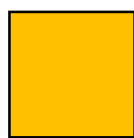
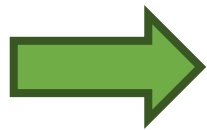
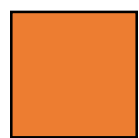
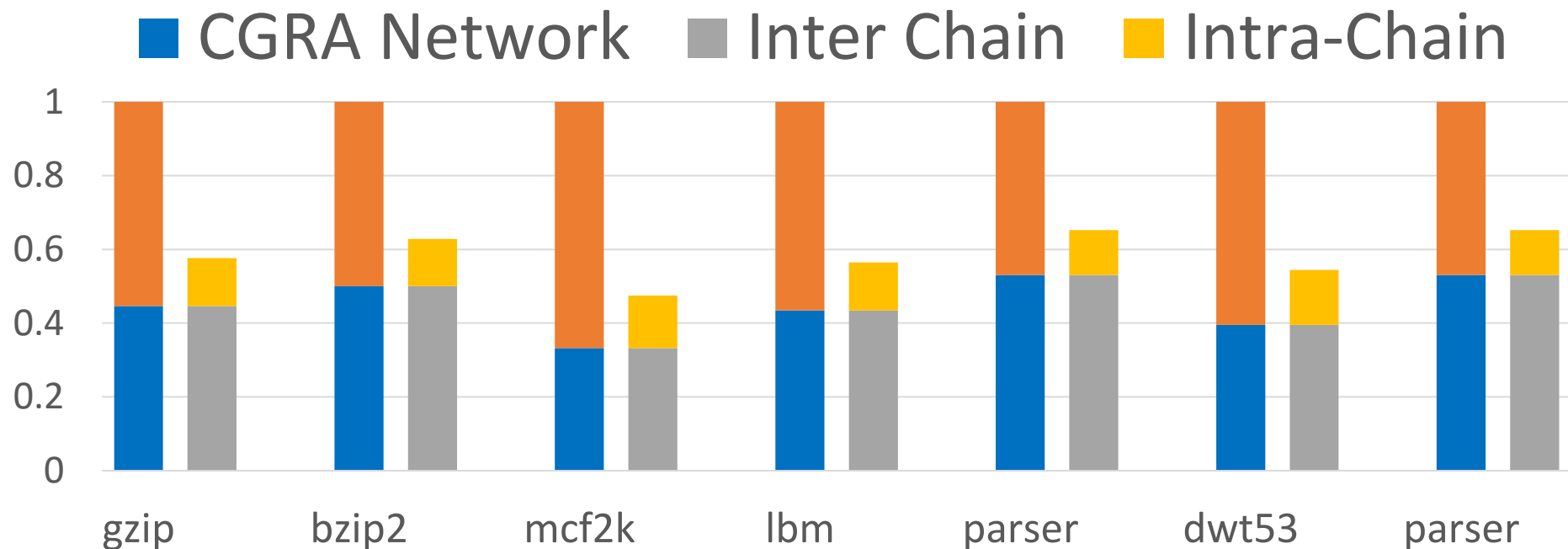


40-60% communication  
reduction

# Multi-Lane CHAINSAW Execution



# Evaluation – Data movement energy



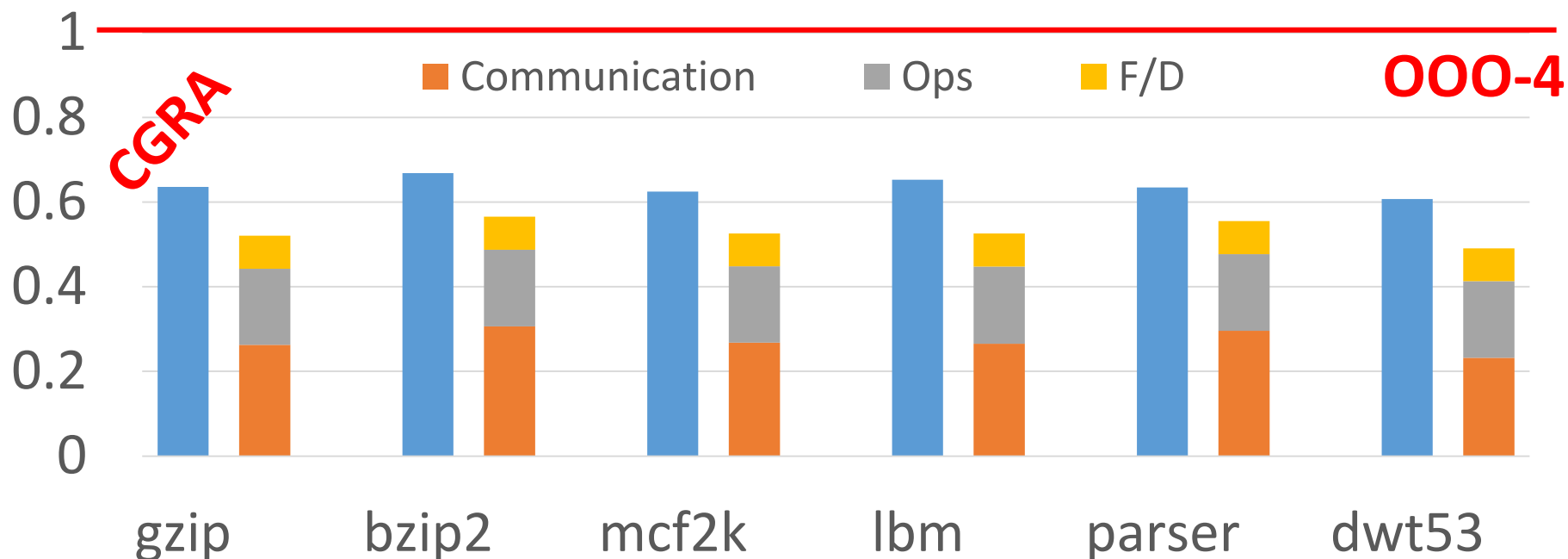
**Reduced energy in  
Chainsaw**

*Chainsaw* internalizes 50%+ of comm.



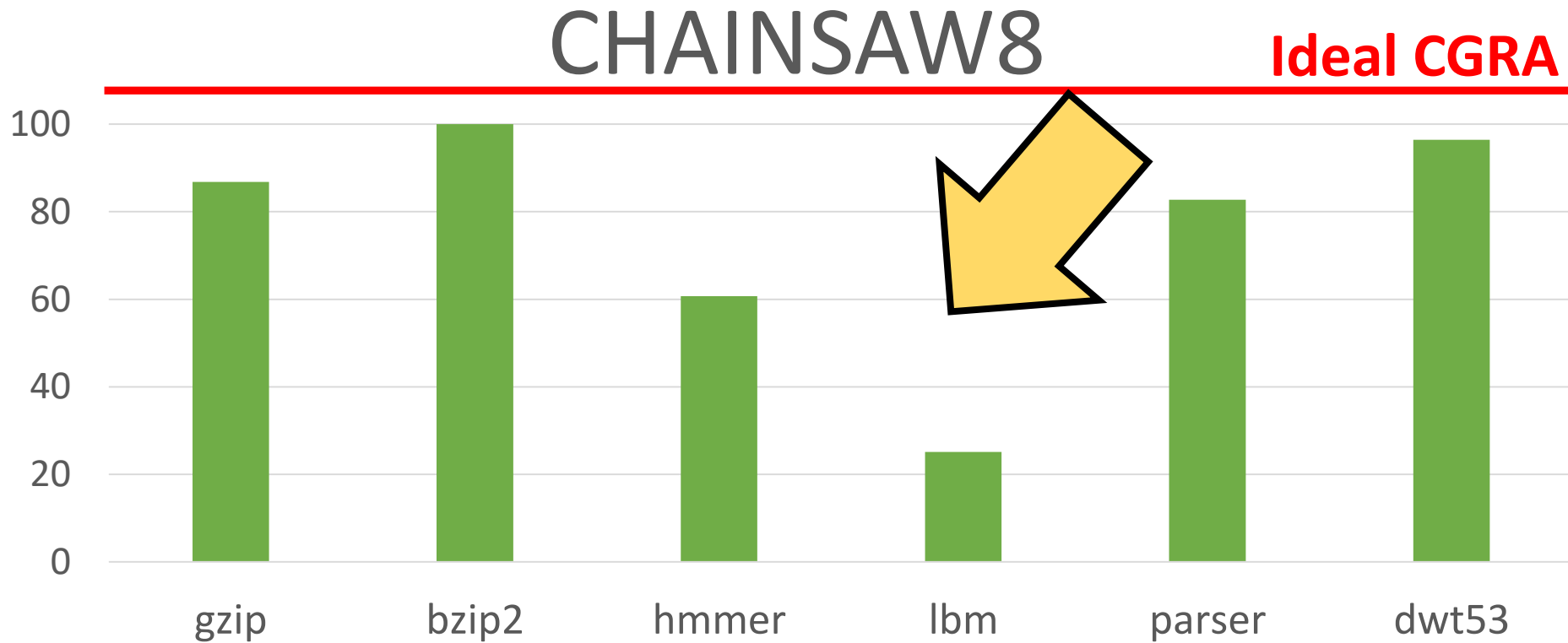
# Evaluation – Dynamic Energy

- *Chainsaw* adds Fetch/Decode cost for dynamic energy
- **CGRA** network overhead dominate *Chainsaw* F/D cost



13% less than CGRA  
45% less than 4-way 000

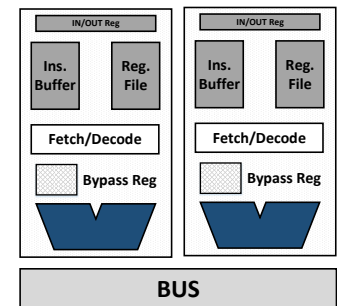
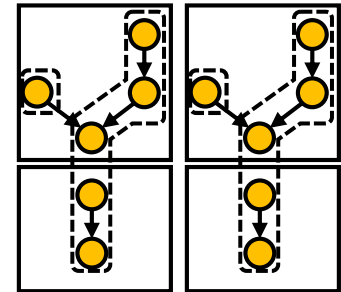
# Evaluation – Performance



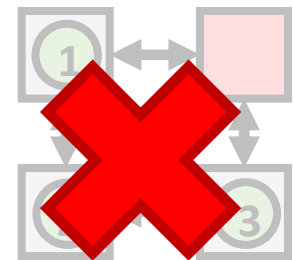
Within 73% of ideal dataflow  
20% better than OOO core

# Chainsaw is a Von-Neumman accelerator

- **Chains** sequentially dependent operations.
- **Chainsaw Accelerator:**
  - Target control free regions
  - Reuse functional units
  - Reduce communication energy
  - Lane based architecture



Energy < CGRA  
Performance  $\simeq$  CGRA



# Thank you

