

# Data-Flow Deadlock Avoidance for Streaming Applications Mapped on Network-on-Chips

Vittorio Zaccaria  
Politecnico di Milano



# Data-Flow Deadlock Avoidance for Streaming Applications Mapped on Network-on-Chips

Vittorio Zaccaria  
Politecnico di Milano

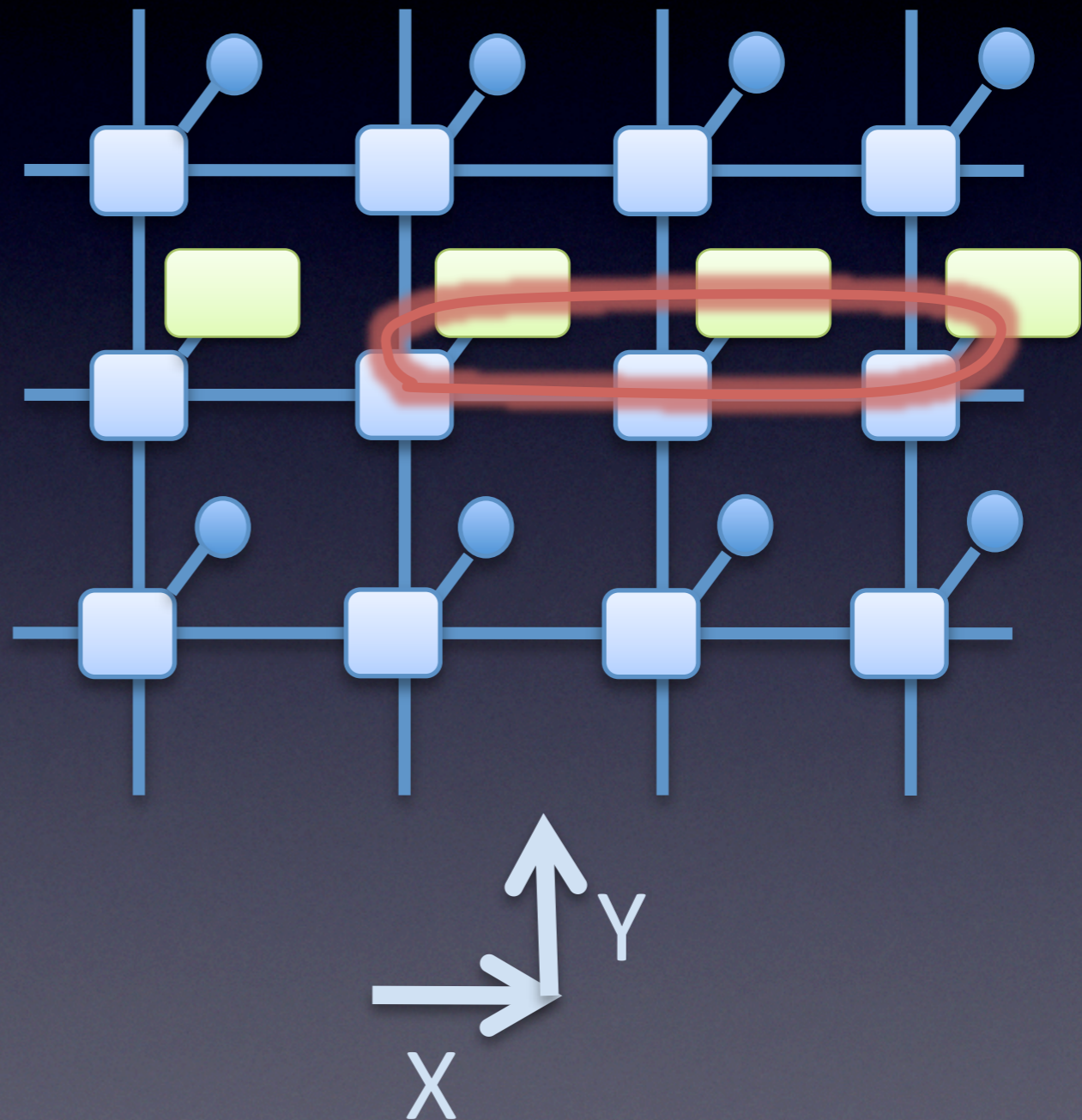
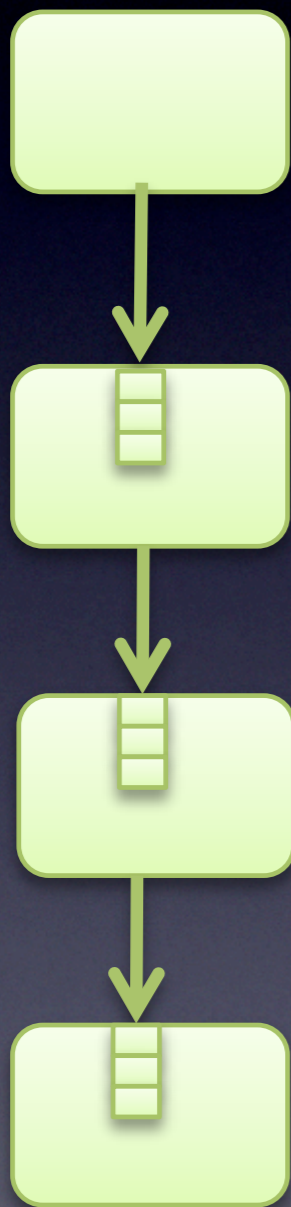


# Outline

- The streaming deadlock problem
- State-of-the art solutions and open questions
- Proposed strategy and future directions



# The streaming deadlock problem





# State-of-the-art

- Significant contributions made by the StreamIt group ['02, '05, '06]
- Communication scheduler statically orchestrates and serializes the communication
- Filter mapping avoids that streams intersect within the NoC topology



# Open questions

- What is the overhead due to the communication scheduling?
- Is it scalable and generalizable?
- How unpredictable/dynamic scenarios are managed?
- What happens when the compiler could not even access physical information about the actual NoC fabric?



# Proposed strategy

- Provide the missing synergy between the OS, the streaming compiler and the architecture.
- Virtualize the architectural streaming resources at the OS level.
- Information about the actual data-flow to be passed from the compiler to the OS.
- Run-time mapping and optimization



# Long term benefits

- Wide-range of run-time optimization possibilities for:
  - Safe communication
  - Ad-hoc run-time program optimization for performance and power consumption
    - Dynamic filter mapping
    - Run-time stream program transformations based on IRs



Questions?