ANDRESTA: AN AUTOMATED NOC-BASED DESIGN FLOW FOR REAL-TIME STREAMING APPLICATIONS

Mahdi Abbaszadeh, Mohammadmahdi Mazraeli, Seyed-Hossein Attarzadeh-Niaki, Dara Rahmati
ANDRESTA

Mohammadmahdi Mazraeli

Bachelor’s Student in Computer Engineering at Shahid Beheshti University
ANDRESTA

- BUS or NoC
- Real-Time Constraints
- Hardware & Software Detailed Knowledge
- Time-Consuming
- Partitioning the Design Space Exploration (DSE) into Sub-Problems
ANDRESTA

Design flow

Constraints (e.g. deadline, throughput, etc.)

Application behavior

Hardware specification

Objective: (e.g. area, power, performance, etc.)

Output design
https://github.com/SBU-CPS-Lab/ANDRESTA
• Synchronous Data Flow (SDF) Graph
• Actor’s High-Level Functions
  • Described By C Language
• Application Representation File
  • Described By an XML
• Platform CP Model
  • Described by an XML
  • Including:
    • # of Processors
    • Local Memory Capacity
    • NoC’s Specification
    • ...
• Constraint Programming
  • Jointly Explore the Design Spaces
  • Declarative
  • Separate Problem Model from Solver Model
• Constraint Satisfaction Problem
• Constraint Optimization Problem
  • Cost Function, Reward Function
Mapping streaming applications on multiprocessors with time-division-multiplexed network-on-chip

Usman Mazhar Mirza*, Flavius Gruian, Krzysztof Kuchcinski

Department of Computer Science, Lund University, 221 00 Lund, Sweden
X-Y Routing

Source

Destination

a b c d

e f g h

i j k l

m n o p
• Cost function can be:
  • Number of Processors
  • Communication Cost
  • Combination of Them

• Cost = $C_1 \times \text{#Processors} + C_2 \times \text{CommunicationCost}$
ANDRESTA
ANDRESTA

• Intermediate Representation
  • Application behavior information
  • Hardware specification
  • Mapping result

• Recipe Generator Engine
  • Python

• Hardware
SOFTWARE

• Actor’s Function (scheduled in a self-timed manner)

HAL

• User interface to communicate in network through each processor’s pair of FIFOs

• Blocking send and receive functions

• Configurable message creator functions
Experimental Results

- Case 1

Cost = $C_1 \times \#\text{Processors} + C_2 \times \text{CommunicationCost}$

$C_1 = C_2$

Desired Throughput = 4 ips  
Achieved Throughput = 6.99 ips

(ips = iteration per second)
Experimental Results

- Case 2

\[
\text{Cost} = C_1 \times \#\text{Processors} + C_2 \times \text{CommunicationCost}
\]

\[C_1 = 1, \quad C_2 = 10\]

Desired Throughput = 4 ips  
Achieved Throughput = 7.26 ips  
(ips = iteration per second)
Conclusion

• A Fully Automated NoC-Based Design Flow for Real-Time Applications on FPGA

• Joint Exploration of the System Synthesis and Platform-Tuning Design Space for a Global Optimum

• Real-Time Throughput Constraint Guarantee Exploiting a CP Solver and an NoC

• Demonstrate the Proposed Design Flow in Action Using a Real-World Case Study
Future works

• Other Routing Algorithms
• Encode NoC’s Performance Bound Computation Into a CP Model
• Support Other Models of Computation
• Support Multiple Applications Simultaneously
• Support Mixed-Critical Systems
Thanks for paying attention!