A Methodology for Constraint-Driven Synthesis of On-Chip Communications

Pinto, Carloni, and Sangiovanni-Vincentelli

Discussion session – EE 249
Behrooz Shahsavari

Outline

- Overview
- Methodology and its representation
- Formulation of the optimization problem
- Application to Network-on-chip synthesis

Overview

- Methodology and an optimization framework for the synthesis of on-chip communication through the assembly of components from a target library.
- Library:
  - Models for functionality, cost, and performance of each element
  - composition rules
- Mathematical framework to model communication at different levels of abstraction
  - point-to-point input specification
  - library elements
  - final implementation

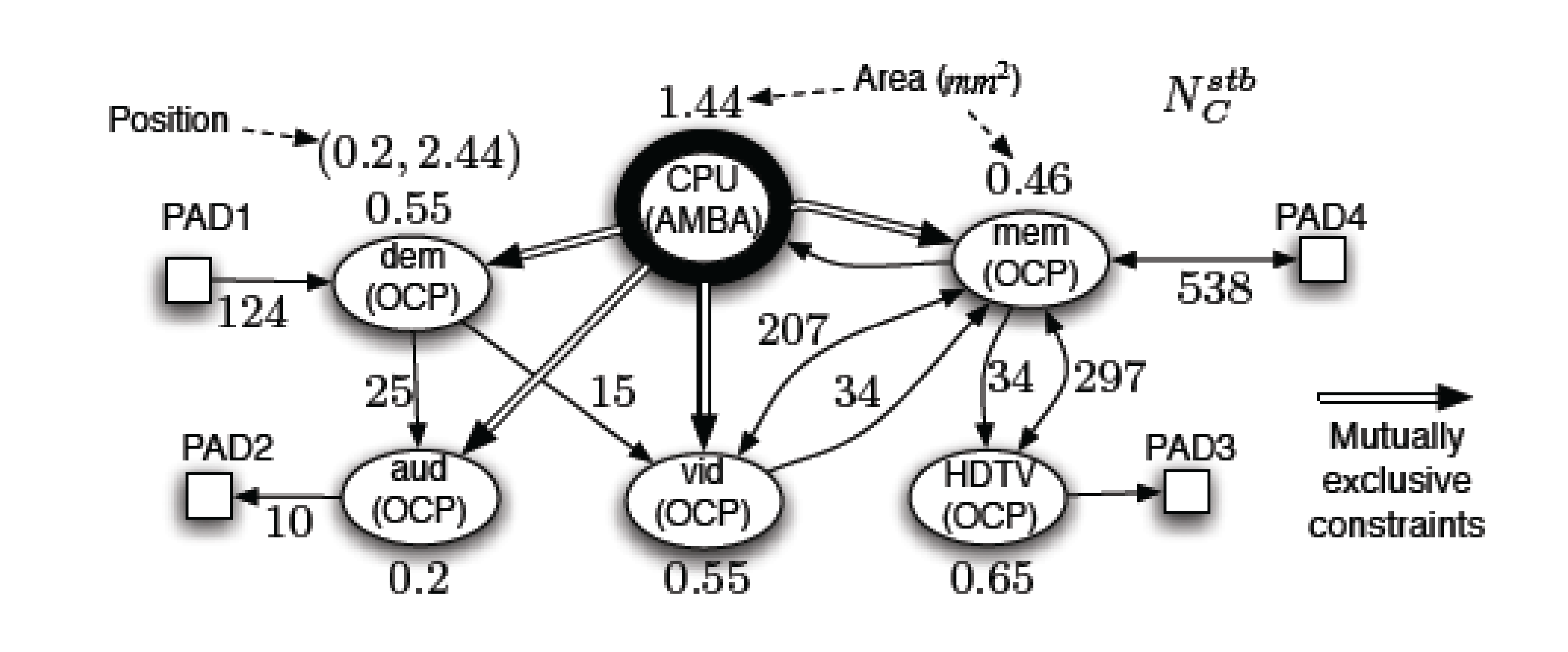
THE METHODOLOGY AND ITS MATHEMATICAL REPRESENTATION

The Methodology

- The general approach is based on Platform-Based Design
- The methodology is recursive
- Platform : a family of admissible solutions
  - set of components together with their compositional rules
- synthesis process
  - select one out of this family (a platform instance)

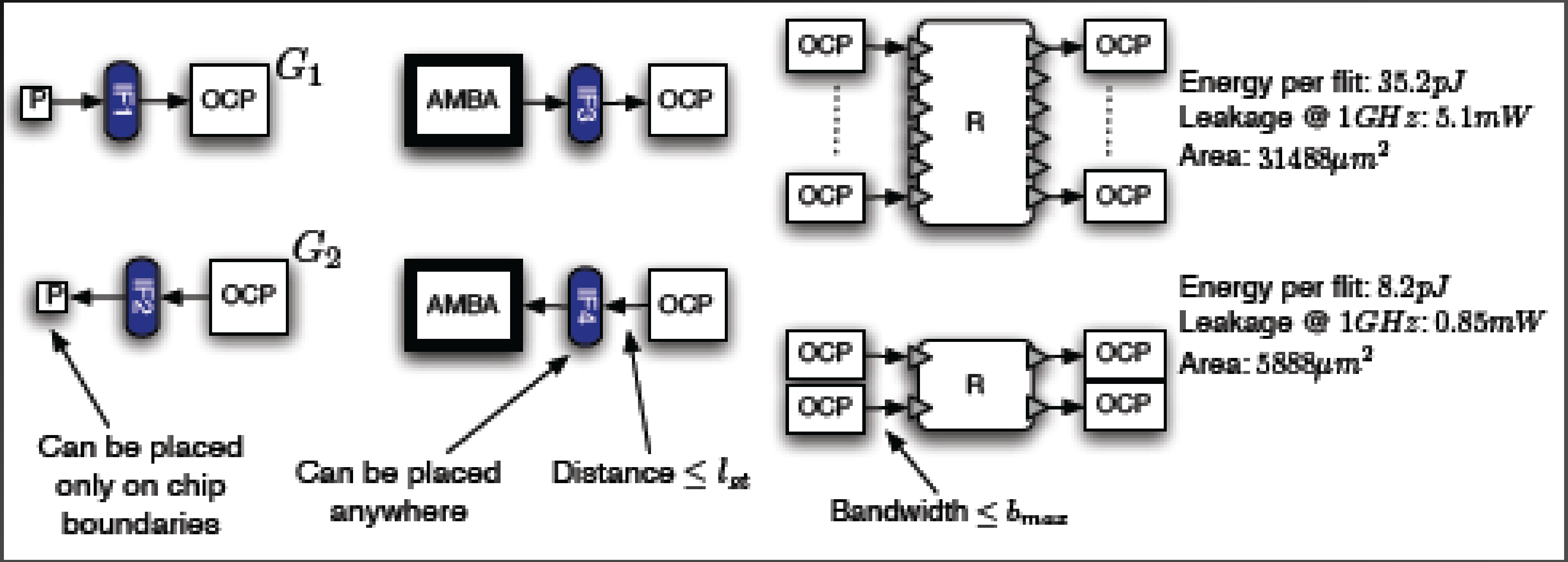
The Methodology

- Characterization:
  - Cost, performance, power, type
- Example: system-level specification of a simplified Set-Top Box



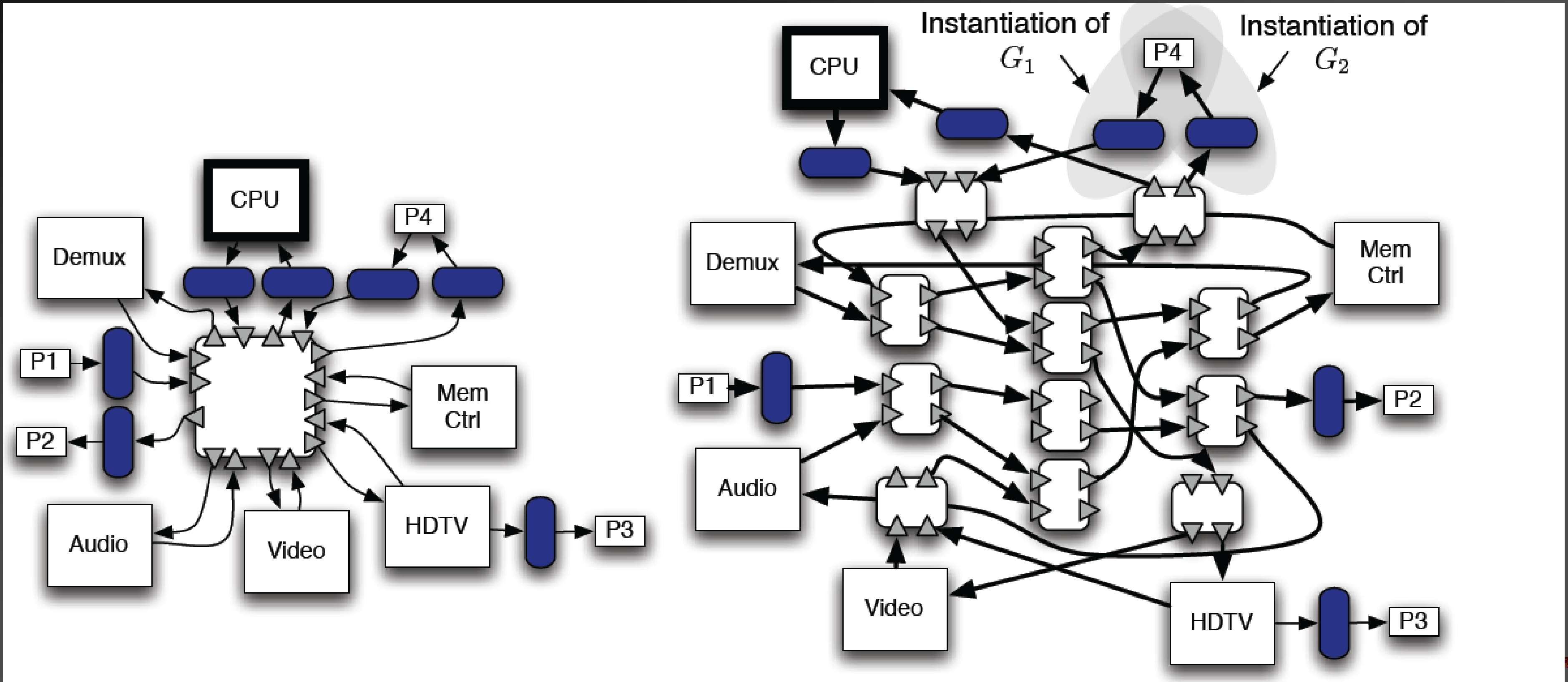
Methodology

- Example: library of predefined on-chip communication components



Methodology

- communication structure
  - instantiating communication templates (i.e. components from the library) and composing them.



Basic Definitions

- Communication structure
  - components with associated quantities
- Quantity q takes on values from a domain D_q
  - $\preceq_q$
  - $\perp$

Definition 1. *A* communication structure *is a tuple* $N(\mathcal{C}, \mathbf{q}, L)$ *where* $\mathcal{C} = \{c_1, \ldots, c_n\}$ *is a set of components,* $\mathbf{q} = (q_1, \ldots, q_k)$ *is a vector of quantities, and* $L \subseteq [\mathcal{C} \to D_\mathbf{q}]$ *is a set of communication configurations. Set* $\mathcal{C}$ *is partitioned into the set of nodes* $V \subseteq U_V$ *and the set of links* $E \subseteq V \times V$.

Basic Definitions

Definition 2. *Given two communication structures* $N_1, N_2 \in \mathcal{G}_{\mathbf{q}}$, $N_1 \leq_{\mathbf{q}} N_2$ *if and only if* $\mathcal{C}_1 \subseteq \mathcal{C}_2$, *and for all* $l_1 \in L_1$ *there exists* $l_2 \in L_2$ *such that for all* $\mathbf{c} \in \mathcal{C}_1$, $l_1(\mathbf{c}) \preceq_{\mathbf{q}} l_2(\mathbf{c})$.

Communication Specification

- Specification of an on-chip communication synthesis problem
  - communication structure $N_c \in G_{q_c}$
  - $q_c = (x, y, a, \tau, b, h)$
- The performance and cost of the network depend on the core positions
  - restrict the possible configurations of a specification by fixing the position of the ports of each core

Communication Structures Instantiation and Composition

- Two operations to allow the incremental design of complex on-chip communications
  - Renaming
  - Parallel composition

Libraries and Platforms

- A platform is the set of all valid compositions that can be obtained by assembling the components from a given communication library
- A communication library L is a collection of communication structures
- The vector of quantities that characterize our platform is $q_p = (x, y, \tau, in, out, \gamma)$

Mapping

- Mapping: for a given platform instance, deriving an implementation of a given specification

- Here, the implementation of a communication specification is a communication structure derived from a platform instance
  - routing of packets and the latency
  - Routing is captured by a quantity ρ called transfer table
  - λ with domain D_λ representing a name attached to each component

Mapping

- An implementation is a communication structure $N_{\mathrm{I}}(\mathrm{C_I}, \mathrm{q_I}, \mathrm{L_I})$ where $q_I = (x, y, \tau, in, out, \rho, b, \gamma, h)$

FORMULATION OF THE OPTIMIZATION PROBLEM

Objective

- Find an implementation N_I that minimizes a given cost function $F : G_{qI} \rightarrow R_+$
  - Cost function is montonic: $N_1 \leq_{qI} N2 \Rightarrow F(N1) \leq_{qI} F(N2)$

$$
\begin{aligned}
\mathrm{PR1}(N_P): \quad & \min_{C_I, L_I} && F(N_I) & \\
& \textit{subject to} && N_C \leq_{\mathbf{q}_C} \Pi(N_I), & (1) \\
& && \Psi(N_I) \in \langle \mathcal{L} \rangle & (2) \\
& && \Psi(N_I) \leq_{\mathbf{q}_P} N_P & (3) \\
& && (C_I, l_I) \in \mathcal{R}_I, \ \forall l_I \in L_I & (4)
\end{aligned}
$$

Optimization

- Let Alg be a hypothetical algorithm that solves problem $PR1$ exactly. Given a library L, platform $\langle L \rangle$ can be explored by using Alg to solve problem $PR1$ for each $N_p \in \langle L \rangle$

Lemma 1. *Let N_C be a specification, $N_{P,1}$ and $N_{P,2}$ two platform instances such that $N_{P,1} \leq_{qP} N_{P,2}$. Let $N^*_{I,1}$ and $N^*_{I,2}$ be the implementations found by Alg for platform instances $N_{P,1}$ and $N_{P,2}$, respectively. Then $F(N^*_{I,2}) \leq F(N^*_{I,1})$.*

APPLICATION TO NETWORK-ON-CHIP SYNTHESIS

The Communication Library and the Composition Rules

- The nodes of our library are routers and network interfaces
- Two important composition rules are considered:
  - At the platform level, rule R_P allows only communication structures
  - At the implementation level, rule R_I allows only deadlock-free communication structures

Solution to the Optimization Problem

- Linearize the problem and solve it using Integer Linear Programming
  - # of variables becomes very large
  - some composition rules cannot be included in the ILP
- A heuristic approach
  - Structure of the Algorithm
  - The FindPath procedure