System-Level Design Space Exploration for HMPSoC

Luigi Pomante
luigi.pomante@univaq.it

University of L’Aquila – ITALY
Center of Excellence DEWS
Overview

• Introduction
• System-Level Design Flow
• Target Architecture
• Application Specification
  – Internal Models
• Design Space Exploration
  – Exploration Strategy
  – System-Level Metrics
• Conclusions
Overview

• **Introduction**
• System-Level Design Flow
• Target Architecture
• Application Specification
  – Internal Models
• Design Space Exploration
  – Exploration Strategy
  – System-Level Metrics
• Conclusions
Introduction

• HMPSoC have been recently exploited for a wide range of application domains, for both the embedded and the general purpose products
  – Such systems can include several and different processors, memories, dedicated ICs and a set of interconnections between them

• They are so complex that the design methodology plays a major role in determining the success of a product
Introduction

• For this, in the past few years, a discrete number of research works has focused on system-level co-design of HMPSoC
  – Each of them has proposed a different approach to the design space exploration but all of them always rely on some designer experience to define some aspects of the target architecture
  • In particular, the definition of the communication architecture is often only an input to the design flow (typically imposed by a platform-based approach)
Introduction

• This talk presents a system-level design space exploration strategy that, starting from the application specification and related constraints, would be able to suggest to the designer
  – an HMPSoC reference architecture
  – an HW/SW partitioning of the given application
  – an allocation of the partitioned entities on the proposed HMPSoC

• Moreover, by means of abstract modeling, the approach try to preserve general applicability and feasibility of the proposed solutions
Introduction

• In particular, this talk focuses on
  – *Modeling Issues*
    • with particular emphasis on model of computation and related “internal models” of representation
  – *System-Level Metrics*
    • used to identify suitable HMPSoC architectures by analyzing the application specification
  – *Design Space Exploration*
    • by means of a genetic algorithm that exploits metrics and some profiling/estimations
      – The approach has been partially validated by checking the consistency with respect to some case studies
Overview

• Introduction
• **System-Level Design Flow**
• Target Architecture
• Application Specification
  – Internal Models
• Design Space Exploration
  – Exploration Strategy
  – System-Level Metrics
• Conclusions
System-Level Design Flow

High-level Flow

Application Specification

Co-Analysis Co-Estimation

Functional Co-Simulation

System Design Exploration

Partitioning and Architecture Selection

Timing Co-Simulation

Low-level Flow

Processors Class

Architectural Constraints

Scheduling Directives

Timing Constraints

Affinity Timing Size

Profiling Communication

Workload Estimation

Load

Timing

Profiling

Communication
Application Specification

High-level Flow

Co-Analysis Co-Estimation

Functional Co-Simulation

System Design Exploration

Partitioning and Architecture Selection

Timing Co-Simulation

Low-level Flow

Processors Class

Architectural Constraints

Scheduling Directives

Library Data

- Affinity
- Timing
- Size

Profiling Communication

Workload Estimation

- Load

Timing Constraints

Timing Constraints

Low-level Flow
Overview

• Introduction
• System-Level Design Flow
• Target Architecture
• Application Specification
  – Internal Models
• Design Space Exploration
  – Exploration Strategy
  – System-Level Metrics
• Conclusions
Target Architecture

- It is an interconnection of some instances of heterogeneous *Basic Building Block*
  - BBB represent the minimal computation, storage and communication unit in the system
    - Distributed memory architecture

- Each BBB is composed of three main elements
  - The Processing Unit (PU)
  - The Local Memory (LM)
  - The Communication Unit (CU)
    - Number and type of possible instances of interconnection links (IL)
Target Architecture

- Basic Building Block
  - Processing Unit
    - GPP, DSP, [uC]: \{€, L_{MAX}\}
    - AS[I]P: \{€, G_{eq_{MAX}}\}
  - Local Memory
    - RAM, ROM: \{€, KB_{MAX}\}
  - Communication Unit
    - Possible Interconnection links
      - IL_{i}:\{BW_{MAX}, N_{MIN}, N_{MAX}, €\}
Target Architecture

• Given several instances of BBB and interconnecting them by means of some instances of the available IL it is possible to build a feasible application-specific architecture on which the application can be mapped on

![Architecture Diagram]

- BBB0: DSP0
- BBB1: ASP0
- BBB2: GPP0
- BBB3: GPP1
- BBB4: GPP5
- BBB5: DSP1
Overview

• Introduction
• System-Level Design Flow
• Target Architecture
• **Application Specification**
  – Internal Models
• Design Space Exploration
  – Exploration Strategy
  – System-Level Metrics
• Conclusions
Application Specification

• System-Level Model of Computation
  – *Communicating Sequential Processes*
    • e.g.

![Diagram of communicating processes](image)
Application Specification

• System-Level Specification Languages
  – Some *homogeneous* languages able to support CSP
    • OCCAM
    • HandelC
    • SystemC
    • Simulink
    • …
Application Specification

• System-Level Internal Models
  – Different levels of details for IM
    • Statement-level IM
      – Dependent on specification languages and parsers
    • Procedure-level IM
      – Independent from specification languages
Application Specification

• System-Level Internal Models
  – The procedure-level internal model used in this work during the design space exploration to represent the CSP is called *Procedural Interaction Graph (PING)*

  – The PING is
    • based on the well-known *Procedural Call Graph*
    • able to capture needed information from imperative, possibly *object-oriented*, specification
Application Specification

- System-Level Internal Models
  - PING derived from a CSP
    - e.g.
Overview

• Introduction
• System-Level Design Flow
• Target Architecture
• Application Specification
  – Internal Models
• Design Space Exploration
  – Exploration Strategy
  – System-Level Metrics
• Conclusions
Design Space Exploration

- The goal of DSE is the automatic identification of both the architecture and the mapping that optimizes a given CF.
Design Space Exploration

- Each procedure in the PING is supposed to be annotated by different metrics
  - Load imposed to a single GPP
    - \( l_i \)
  - Bandwidth needed to communicate with other procedures
    - \( b_i \)
  - Size
    - \( s_i \): KB or \( G_{eq} \)
  - Affinity towards GPP, DSP and ASP
    - \( a_i \)
Design Space Exploration

- System-Level Internal Models
  - Annotated PING derived from CSP
    - e.g.
Design Space Exploration

• 1\textsuperscript{st} phase
  – Starting from PING to determine number and type of BBB/PU…
    • Goals
      – Minimize the cost of the set of BBB/PU
        » Max number for each kind of PU could be provided by the designer
      – Exploit the potential parallelism expressed in the PING
      – Keep the load of each PU near but under its \( L_{MAX} \)
      – Minimize communications between different BBB/PU
      – Keep the used size near but under \( KB_{MAX} \) or \( G_{eq\_MAX} \)
      – Exploit affinity between BBB/PU and the procedures
Design Space Exploration

• 1st phase
  – …minimizing a cost function by means of a genetic approach
Design Space Exploration

- **System-Level Metrics (1st phase)**
  - *Affinity Index* ($I_A$) [0, 1]
    - Matching between the functionalities and the processing elements on which they have been allocated
  - *Load Indexes* ($I_{Lsw}$, $I_{Lhw}$) [0, 1]
    - Balancing of the workload over the available HW and SW processing elements
  - *Communication Index* ($I_C$) [0, 1]
    - Exchanged data size between functionalities allocated on different processing elements
  - *Physical Cost Index* ($I_\$, ) [0, 1]
    - Cost of the solution with respect to the most expensive one
Design Space Exploration

• System-Level Metrics (1\textsuperscript{st} phase)
  – By combining the metrics a cost function has been built to compare different solutions

\[ CF = w_A \cdot I_A + w_{Lsw} \cdot I_{Lsw} + w_{Lhw} \cdot I_{Lhw} + w_c \cdot I_c + w_s \cdot I_s \]

• Affinity and load parameters
  – Tend mainly to separate the functionalities to balance the load and exploit the processing elements features

• Communications and physical cost
  – Tend to keep together the functionalities to minimize the number of processing elements
Design Space Exploration

• 2nd phase
  – From the results of the 1st phase and the PING it is possible to build a BBB Interaction Graph (BING)
    • The BING is a model used to represent the partial system at the end of the first phase
      – Each edge will present a proper $B=f(b_i)$
Design Space Exploration

• 2nd phase
  – Starting from BING to determine number and type of IL between PU…
    • Goals
      – Minimize cost of IL$_i$
        ▷ Max number of instances for each IL can be specified by the designer
      – Keep the bandwidth of each IL$_i$ under but near $BW_{MAX}$
      – Keep the number of executors of each IL$_i$ under but near $N_{MAX}$ (and > $N_{MIN}$)
      – Satisfy eventual constraints on the latency
      – Keeping feasibility while respecting CU characterization
Design Space Exploration

• 2\textsuperscript{nd} phase
  - …minimizing a cost function by means of a genetic approach
Design Space Exploration

- **System-Level Metrics (2\textsuperscript{nd} phase)**
  - *Saturation Index* ($I_B$) [0, 1]
    - Respect of max bandwidth offered by the IL
  - *Exploitation Index* ($I_E$) [0, 1]
    - Respect min/max number of BBB that can use a single IL instance
  - *Physical Cost Index* ($I_\epsilon$) [0, 1]
    - Cost with respect to the expensive solution
  - *Feasibility Index* ($I_F$) [0, 1]
    - A pair of CU should be able to manage at least a common IL in order to allow the related BBB to directly communicate
Design Space Exploration

• System-Level Metrics (2\textsuperscript{nd} phase)
  – CU characterization for feasibility
    • e.g.

<table>
<thead>
<tr>
<th></th>
<th>IL0</th>
<th>IL1</th>
<th>IL2</th>
<th>IL3</th>
<th>IL4</th>
</tr>
</thead>
<tbody>
<tr>
<td>BBB0-CU</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>BBB1-CU</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>BBB2-CU</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>BBB3-CU</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BBB4-CU</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

  – Cost Function
    • Used to compares different proposed solutions identifying the one that better tradeoffs different parameters

\[
CF = w_B \cdot I_B + w_E \cdot I_E + w_e \cdot I_e + I_f
\]
Design Space Exploration

• Summary

\[ P_1 \rightarrow b_1 \rightarrow b_2 \rightarrow b_3 \rightarrow b_4 \rightarrow b_5 \rightarrow b_6 \rightarrow b_7 \rightarrow b_8 \rightarrow b_9 \rightarrow P_8 \]

\[ P_1 \rightarrow P_2 \rightarrow P_3 \rightarrow P_4 \rightarrow P_5 \rightarrow P_6 \rightarrow P_7 \rightarrow P_8 \]

\[ b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8, b_9 \]
Overview

• Introduction
• System-Level Design Flow
• Target Architecture
• Application Specification
  – Internal Models
• Design Space Exploration
  – Exploration Strategy
  – System-Level Metrics
• **Conclusions**
Conclusions

• This talk has presented a System-Level Design Space Exploration strategy for HMPSoC
  – It has presented a methodology able to propose an HW/SW partitioning of the specification, mapping this one onto an automatically selected architecture
    • The methodology has been only partially validated

• The experimental results are encouraging and justify further research efforts in this direction may be also thanks to your comments…