Modeling and Synthesis of the Network in Distributed Embedded Systems

Emad Ebeid
University of Verona, Italy
Emad.Ebeid@univr.it

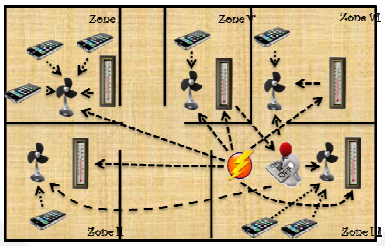
Advisors:
Franco Fummi,
Davide Quaglia
University of Verona, Italy

Outline

- Introduction & Problem statement
- Objectives
- Proposed Methodology
  - Modeling
  - Synthesis
  - Manipulation and DSE
  - Design validation
- Innovations
- Methodology implementation
- Unified case study
- Conclusions and future research opportunities

Introduction

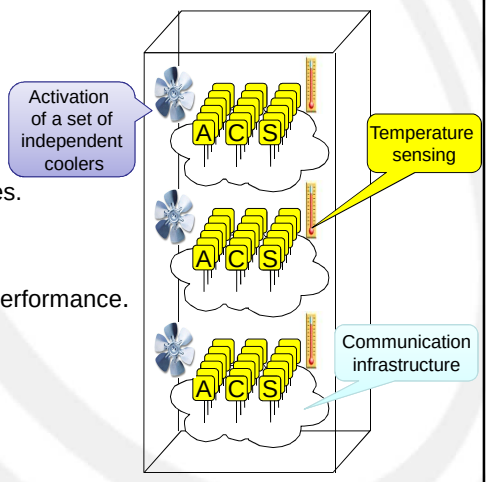
- **D**istributed **E**mbedded **S**ystems (**DES**'s) are distributed applications of Networked Embedded Systems which are special-purpose, resource-constrained nodes interacting together by using:
  - Network interfaces;
  - Standardized protocols;
  - Wired/Wireless Channels.
- **DES** design involves:
  - HW/SW components;
  - Network;
  - Environment.



Introduction (cont'd)

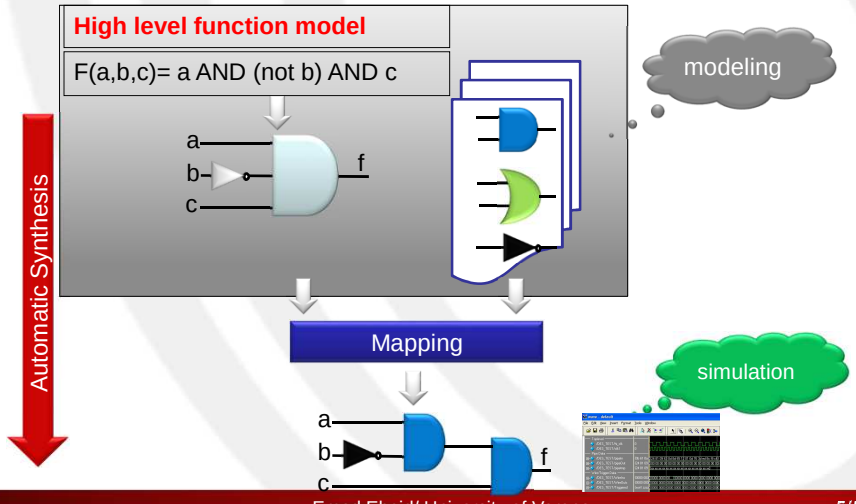
Temperature control of a building

- **Scenario**:
  - Hundreds of concurrent tasks.
  - Heterogeneous tasks.
  - Devices with different capabilities.
  - Wireless and wired channels.
  - Many communication protocols.
  - Nodes position affects system performance.
- **Questions:**
  - How many nodes?
  - How to assign tasks to nodes?
  - Which network protocols?
  - etc.



Introduction (cont'd)

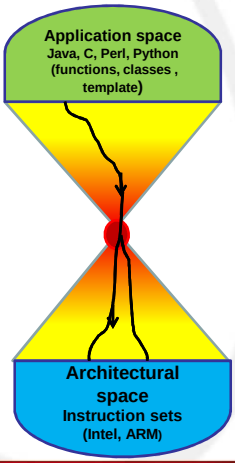
Hardware design:



Introduction (cont'd)

Software development

- Functionality is described with different languages and an automatic process is used to generate assembly code for different target CPU's
- **Modeling** of the functionality: High level languages
- **Automatic synthesis**: Compilers



Problem statement

- **D**esign of DES needs a common representation of different aspects (HW, SW, computation, communication, and environment) and efficient simulation
- **M**ethodology should start from a standard representation of the requirements and the expected behavior of the whole distributed system
- **S**ynthesis and **A**utomatic design-space exploration of the network characteristics of the system (type of channels, protocols and related HW/SW interfaces) are required

Problem statement (cont'd)

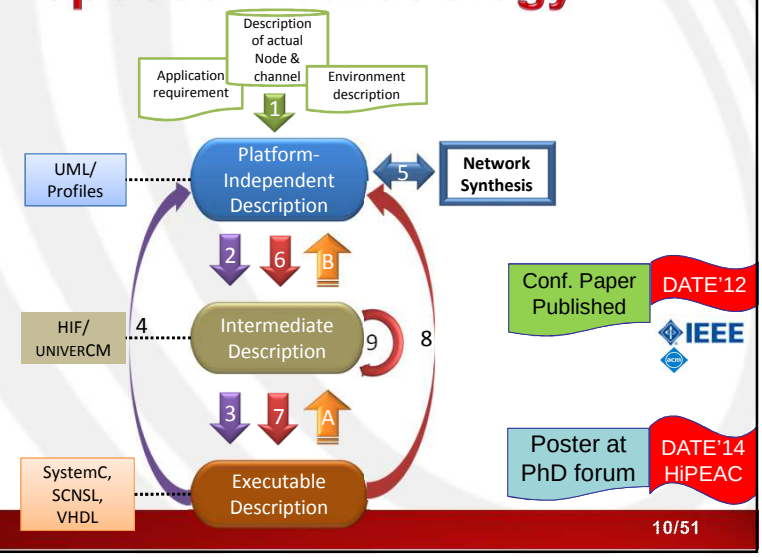
- **S**imulation of the final system must be used to support analysis and synthesis steps
- **P**re-designed and **p**re-verified Intellectual Properties (IPs) should be reused at higher-level of abstraction

Objectives

The thesis aims to provide a design methodology and tools for modeling and synthesis of the network in distributed embedded systems; the methodology should be based on:

- a *common representation framework* (e.g., based on UML with custom profiles, and Heterogeneous Intermediate Format (HIF) to model the different aspects of a DES
- a *mechanism to synthesize* the network from the common representation and to generate efficient simulations

Proposed Methodology



Innovations

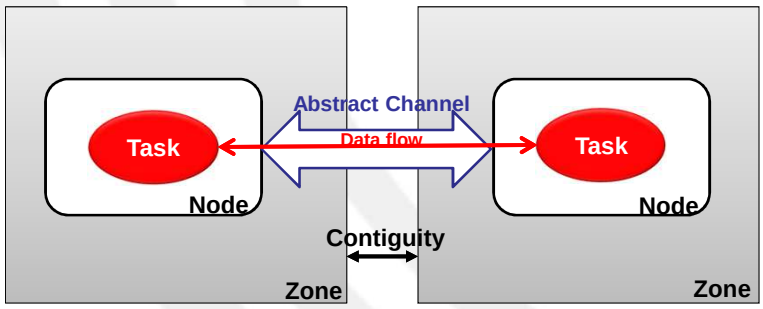
- A design methodology for DES
  - a unique, mathematical model to describe and manipulate DES's
  - a common representation framework to express the different aspects of DES's
- A methodology to model DES aspects
  - a novel UML *Network Profile*
  - a methodology to abstract and integrate heterogeneous IPs
- A technique to synthesize the network in distributed embedded systems
- A comprehensive design space exploration framework

Innovations (cont'd)

- A methodology to automatically validate the high-level models
- The *development* of four tools built upon the HIFSuite framework and an *upgrading* of SCNSL network simulator that were required to implement the proposed methodologies

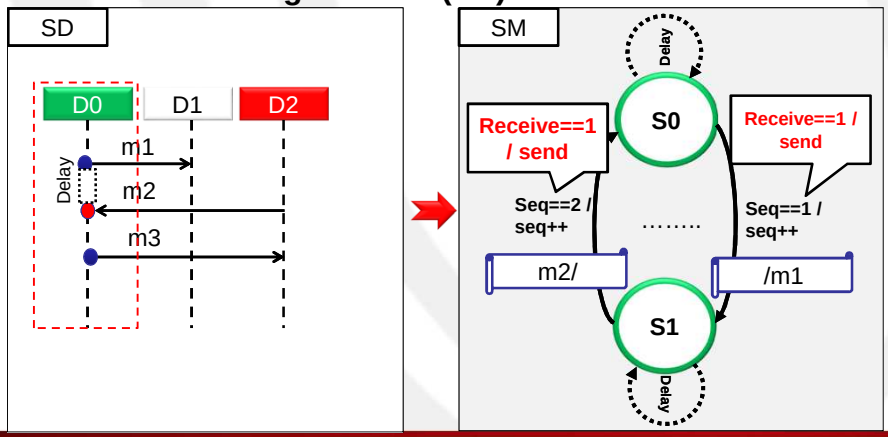
Phase #1: Structural Modeling

Elements of Distributed Embedded Systems



Phase #1: Behavioral Modeling

Behavioral Modeling of tasks (SD)

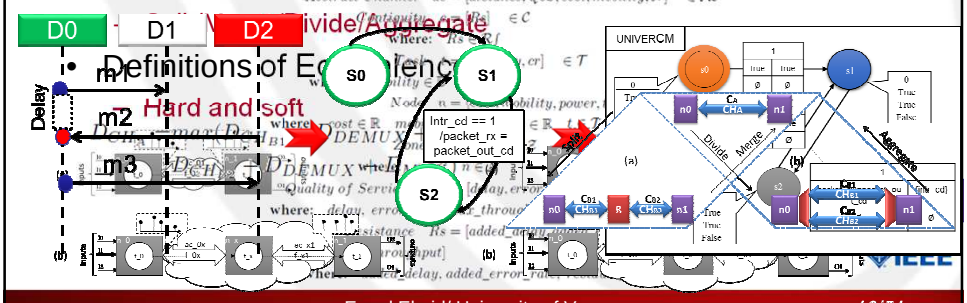


Phase #1: UML/Profiles

- Platform-Independent Models (PIM)
  - UML
- MARTE Profile
- Lack of Formalism
- Lack of semantics
  - Device mobility
  - Network QoS
    - Error rate
  - Environmental modeling
  - etc.

Phase #1: Formalization

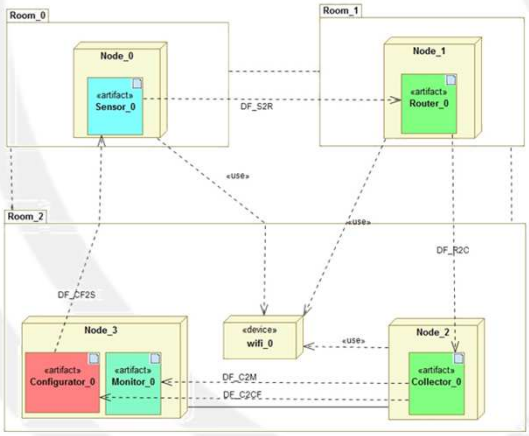
- Formal structural representation
- Formal behavioral representation
  - UNIVERCM
- Formalization of manipulation rules
- Definitions of Equivalence
  - Hard and soft



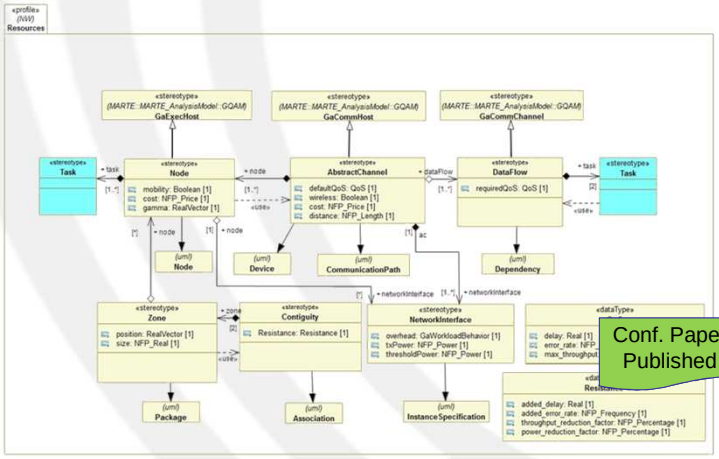
Phase #1: UML Deployment diagram

EXAMPLE

Example of DES



Phase #1: UML *Network Profile*



Conf. Paper Published

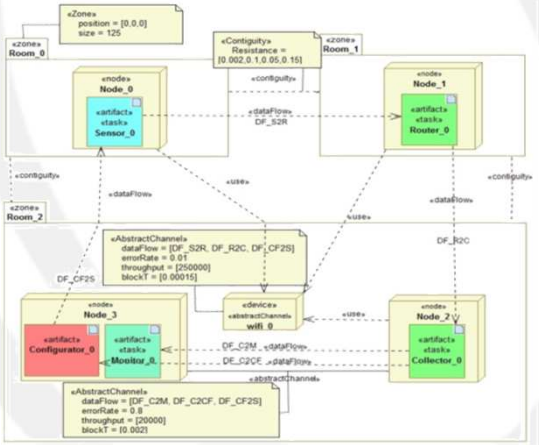
DSD'13

IEEE

Phase #1: UML Deployment diagram + *Network Profile*

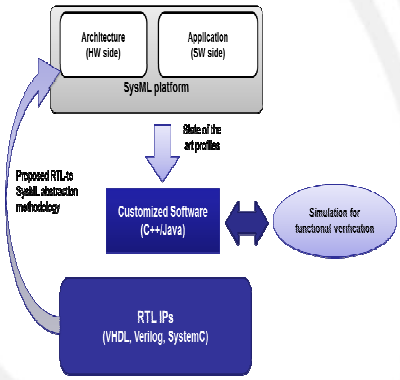
EXAMPLE

Example of DES



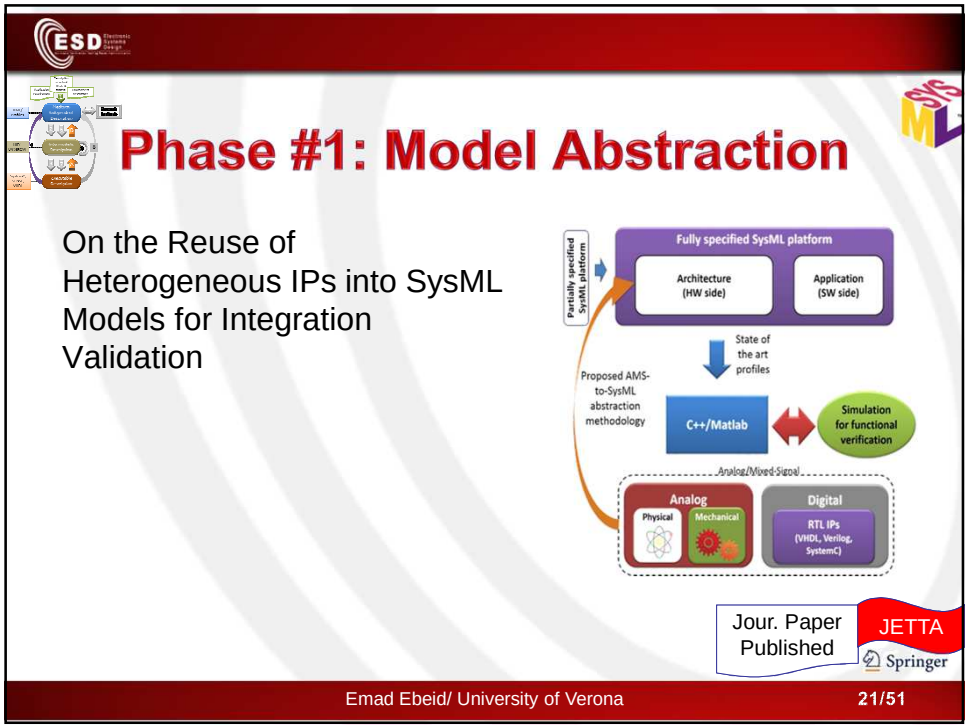
Phase #1: Model Abstraction

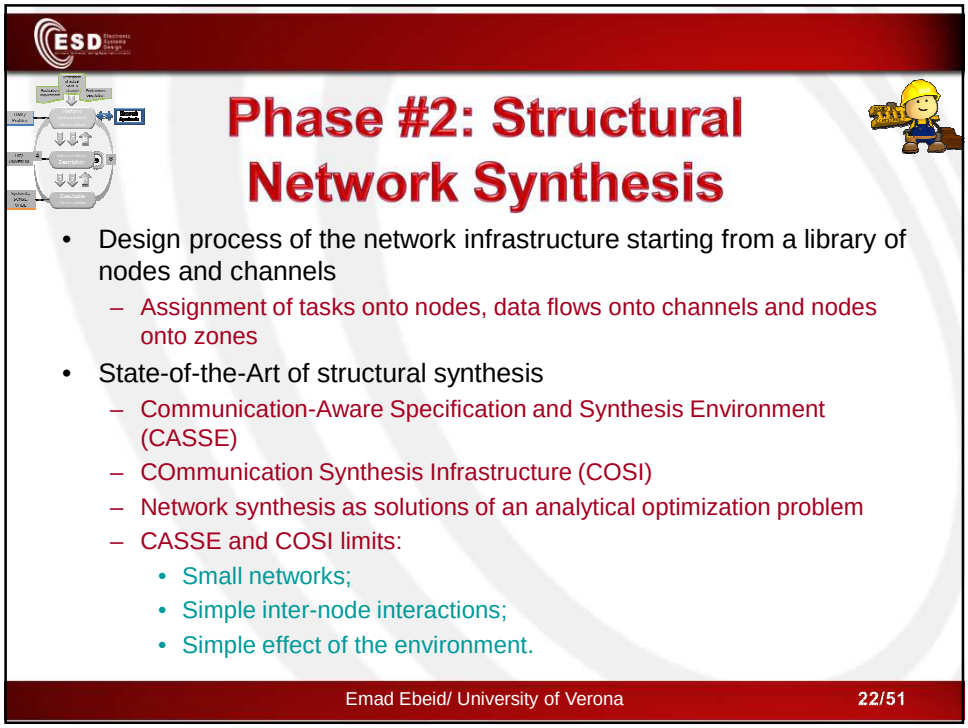
On the Reuse of RTL IPs for SysML Model Generation



Conf. Paper Published MTV'12

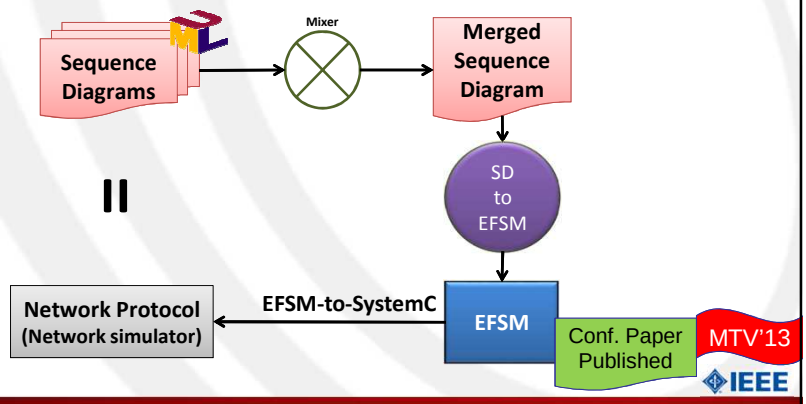
IEEE


ESD
Phase #1: Model Abstraction
On the Reuse of Heterogeneous IPs into SysML Models for Integration Validation
Fully specified SysML platform
Partially specified SysML platform
Architecture (HW side)
Application (SW side)
State of the art profiles
Proposed AMS-to-SysML abstraction methodology
C++/Matlab
Simulation for functional verification
Analog/Mixed-Signal
Analog
Physical
Mechanical
Digital
RTL IPs (VHDL, Verilog, SystemC)
Jour. Paper Published
JETTA
Springer
Emad Ebeid/ University of Verona
21/51


ESD
Phase #2: Structural Network Synthesis
• Design process of the network infrastructure starting from a library of nodes and channels
– Assignment of tasks onto nodes, data flows onto channels and nodes onto zones
• State-of-the-Art of structural synthesis
– Communication-Aware Specification and Synthesis Environment (CASSE)
– COmmunication Synthesis Infrastructure (COSI)
– Network synthesis as solutions of an analytical optimization problem
– CASSE and COSI limits:
• Small networks;
• Simple inter-node interactions;
• Simple effect of the environment.
Emad Ebeid/ University of Verona
22/51

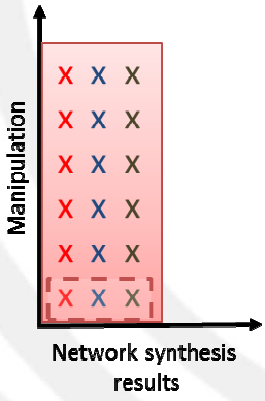
Phase #2: Behavioral Network Synthesis

Protocol Synthesis



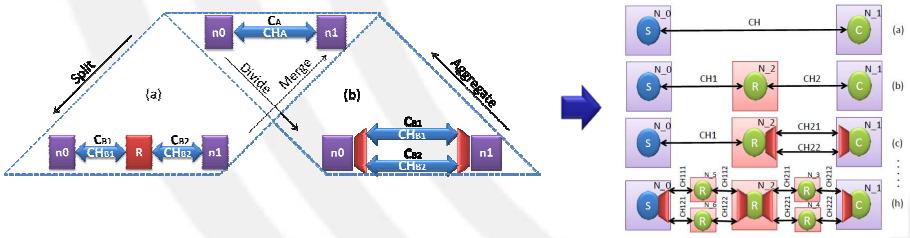
Phase #2: Network Synthesis

Relationship between network synthesis and manipulation:



Phase #2: Manipulation

Structural Manipulation Approach: It aims to help the designer to optimize the synthesized network configuration



Phase #2: Manipulation (cont'd)

Rules for manipulation:

- Mathematical equations to prove equivalence regarding throughput and delay

$$D_{CH_A} = D_{CH_{B1}} + D_{CH_{B2}} + D_R$$

$$C_A = min(C_{B1}, C_{B2})$$

$$\sum (D_{CH_B} + D_R) = D_{CH_A}$$

$$min(C_{B_1}, ..., C_{B_n}) = C_A$$

$$\vdots$$

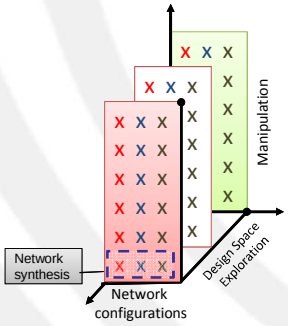
Conf. Paper Published — MTV'13 — IEEE

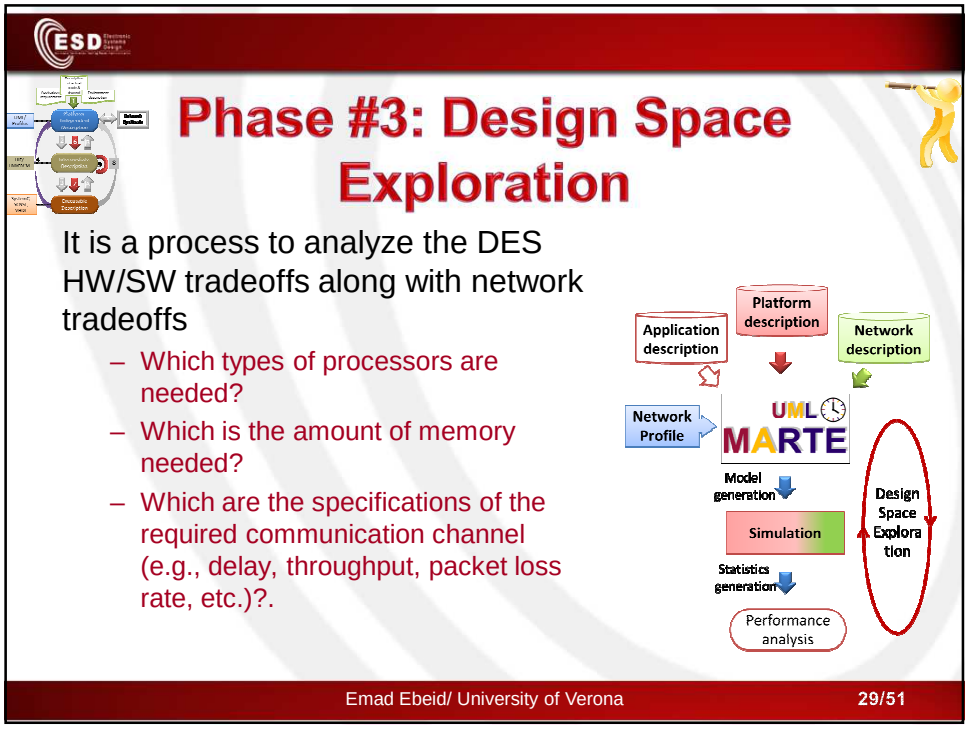
Phase #2: Network Synthesis and Simulation

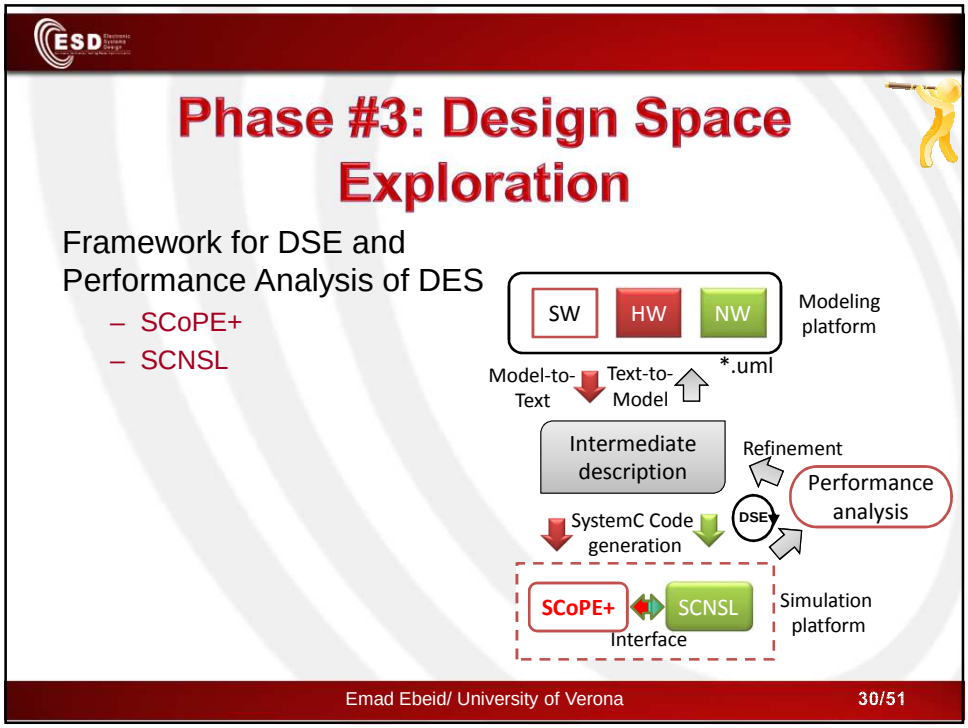
- The analytical framework for Network synthesis does not capture:
  - Complex inter-node interactions (e.g., shared channel overhead)
  - The effect of the surrounding environment
- The optimal solution must be validated through simulation
  - SystemC Network Simulation Library (SCNSL)

Phase #2: Network Synthesis

Relationship between Network synthesis, Manipulation and Design space exploration:




ESD
Phase #3: Design Space Exploration
It is a process to analyze the DES HW/SW tradeoffs along with network tradeoffs
– Which types of processors are needed?
– Which is the amount of memory needed?
– Which are the specifications of the required communication channel (e.g., delay, throughput, packet loss rate, etc.)?.
Application description
Platform description
Network description
Network Profile
UML
MARTE
Model generation
Simulation
Statistics generation
Performance analysis
Design Space Exploration
Emad Ebeid/ University of Verona
29/51

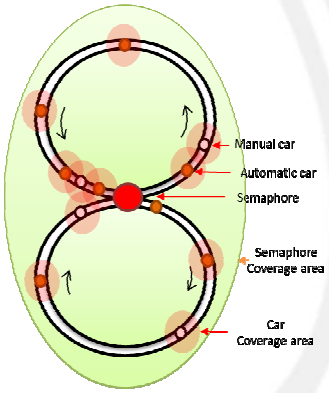

ESD
Phase #3: Design Space Exploration
Framework for DSE and Performance Analysis of DES
– SCoPE+
– SCNSL
SW
HW
NW
Modeling platform
*.uml
Model-to-Text
Text-to-Model
Intermediate description
Refinement
Performance analysis
SystemC Code generation
DSE
SCoPE+
SCNSL
Interface
Simulation platform
Emad Ebeid/ University of Verona
30/51

Phase #3: Design Space Exploration (cont'd)

EXAMPLE

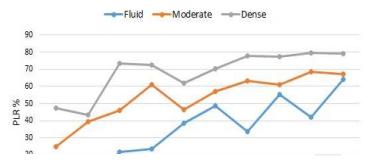
Example

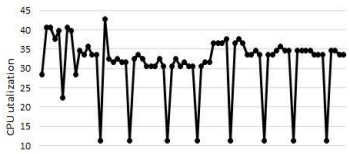
- The case study consists of two types of cars (automatic and manual) and a semaphore.
- Cars are moving in a road which has a figure of 8 shape and the semaphore is located at the middle of the road.

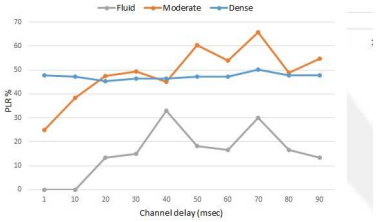


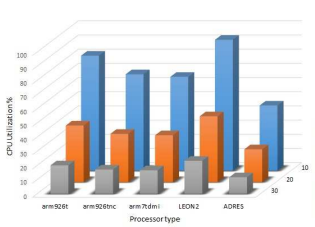
Phase #3: Design Space Exploration (cont'd)

Results:

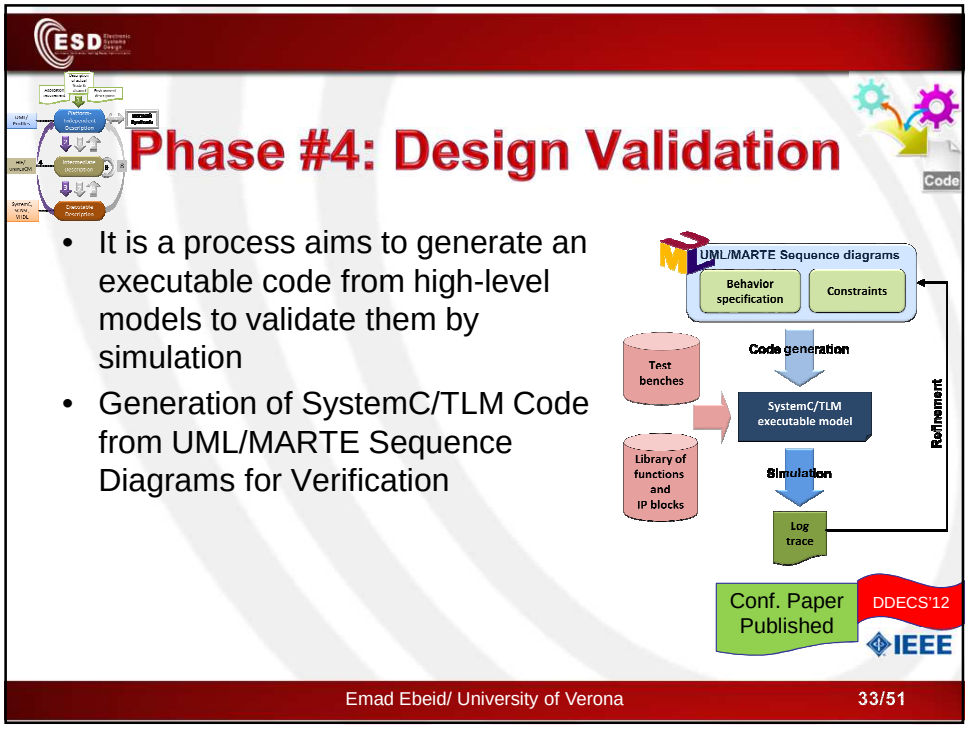


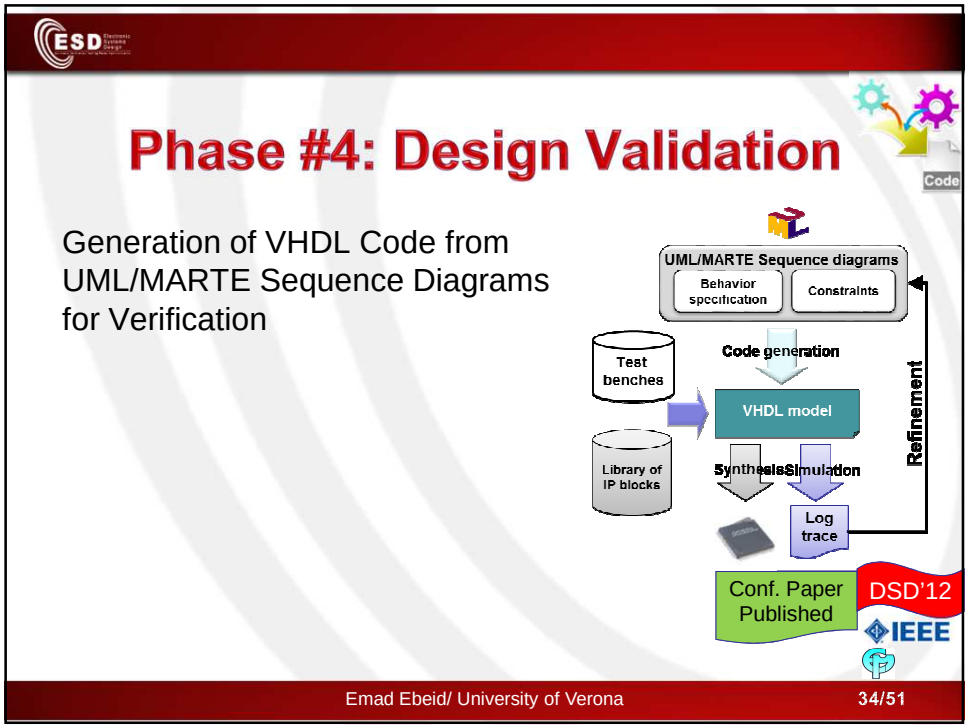






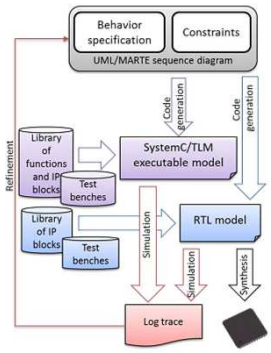
Conf. Paper Published — Rapido'14


ESD
Phase #4: Design Validation
Code
It is a process aims to generate an executable code from high-level models to validate them by simulation
Generation of SystemC/TLM Code from UML/MARTE Sequence Diagrams for Verification
UML/MARTE Sequence diagrams
Behavior specification
Constraints
Test benches
Code generation
SystemC/TLM executable model
Library of functions and IP blocks
Simulation
Log trace
Refinement
Conf. Paper Published
DDECS'12
IEEE
Emad Ebeid/ University of Verona
33/51


ESD
Phase #4: Design Validation
Code
Generation of VHDL Code from UML/MARTE Sequence Diagrams for Verification
UML/MARTE Sequence diagrams
Behavior specification
Constraints
Test benches
Code generation
VHDL model
Library of IP blocks
Synthesis
Simulation
Log trace
Refinement
Conf. Paper Published
DSD'12
IEEE
Emad Ebeid/ University of Verona
34/51

Phase #4: Design Validation

HDL Code Generation From
UML/MARTE Sequence Diagrams
For Verification and Synthesis

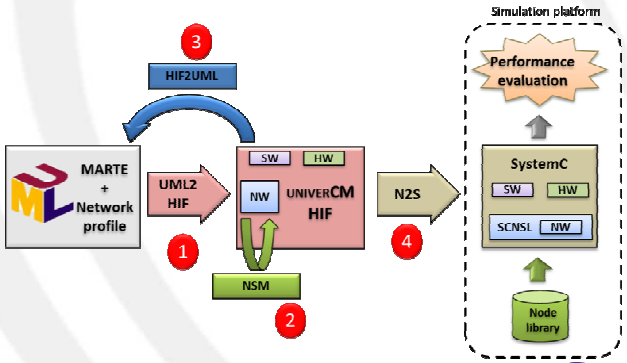


Jour. Paper submitted — DAES — Springer

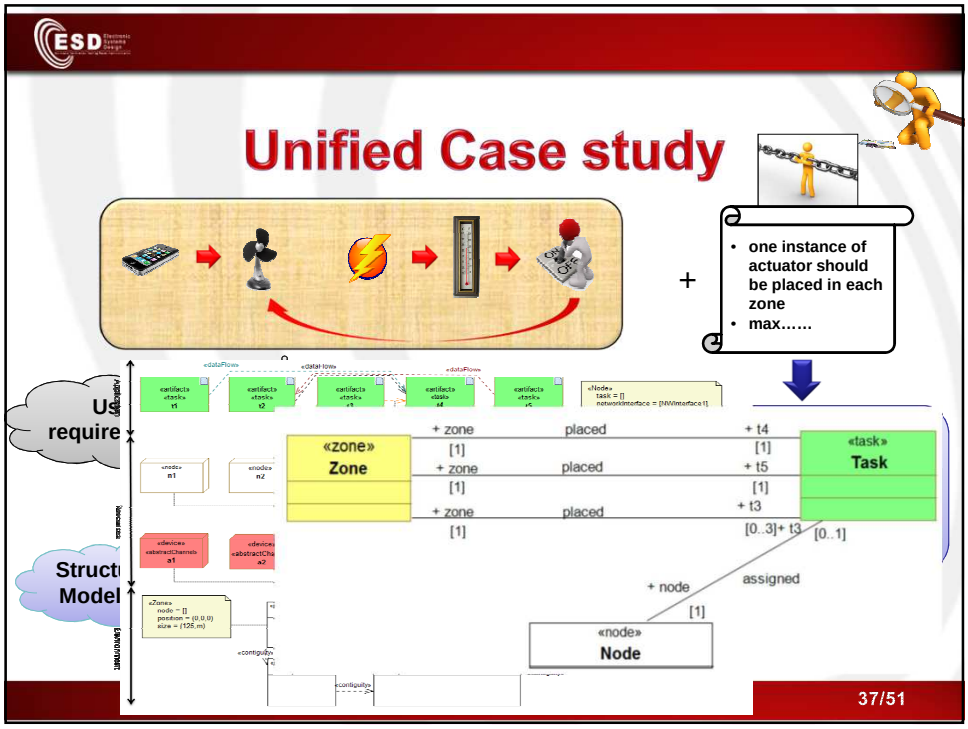
Methodology implementation

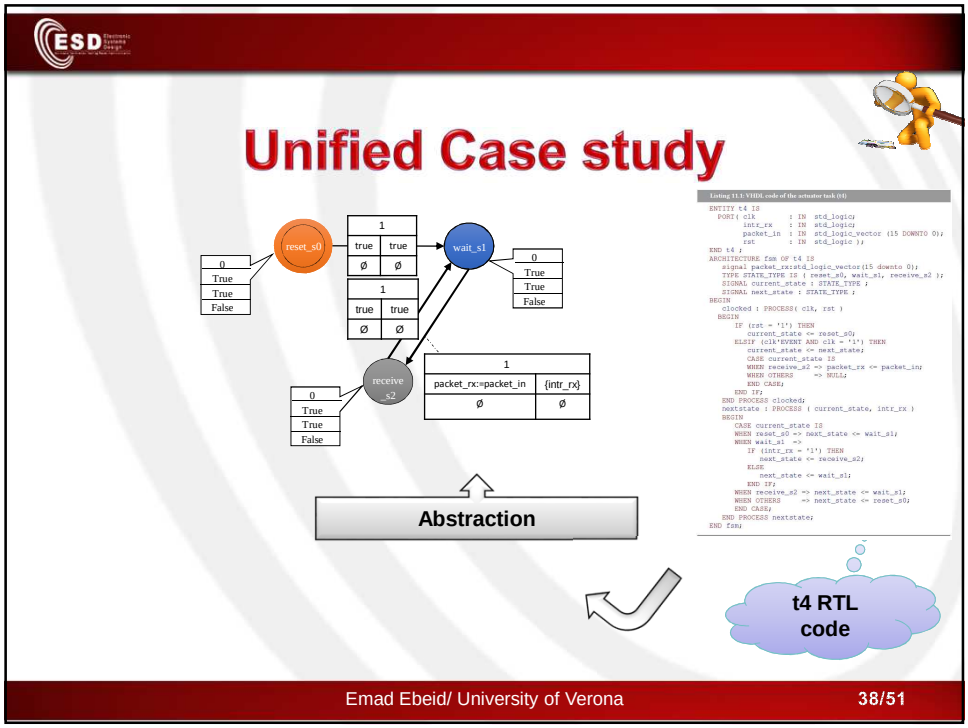
Tools Are:

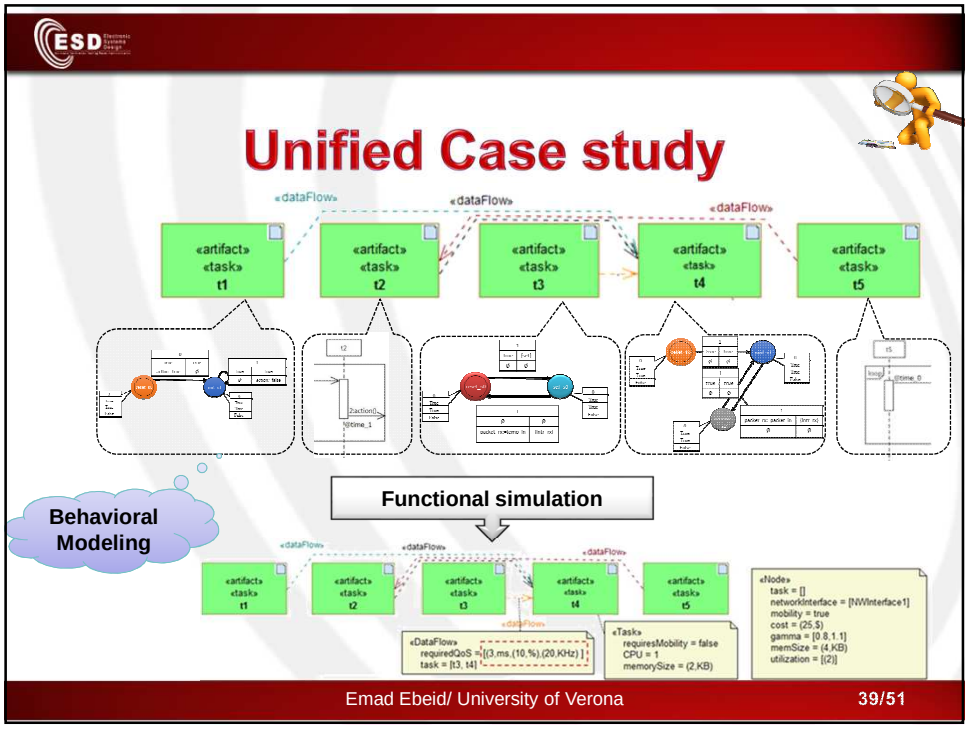
1. UML2HIF
2. NSM
3. HIF2UML
4. N2S

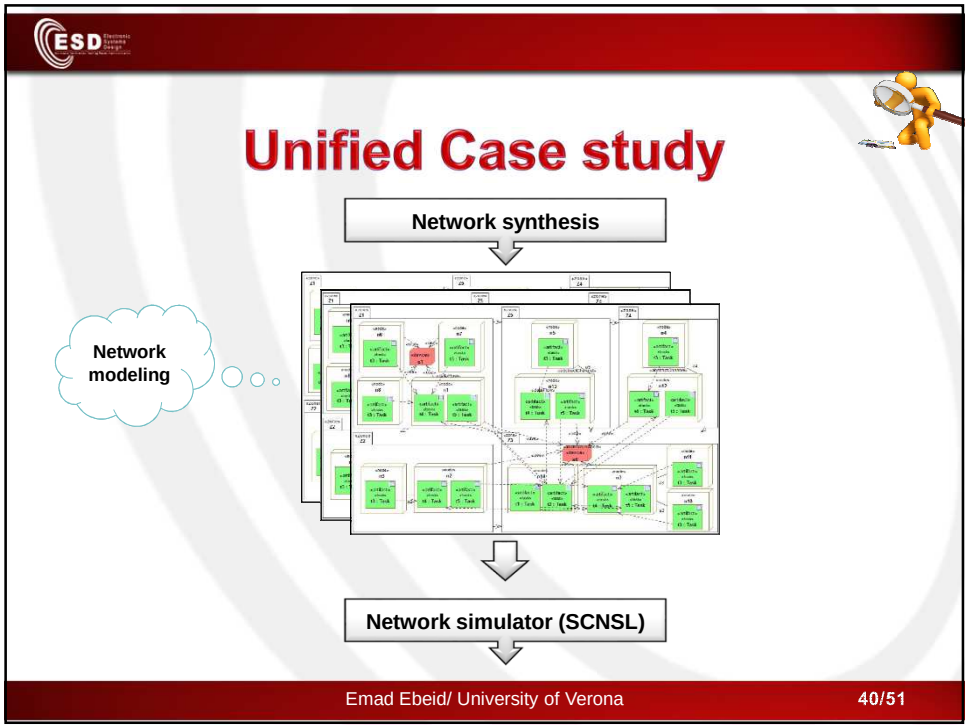


Conf. Paper Published — UKSim'13 — IEEE


Unified Case study
+
one instance of actuator should be placed in each zone
max……
«zone»
Zone
«task»
Task
«node»
Node
+ zone
placed
+ t4
+ t5
+ t3
[1]
[0..3]+ t3
[0..1]
+ node
assigned
37/51

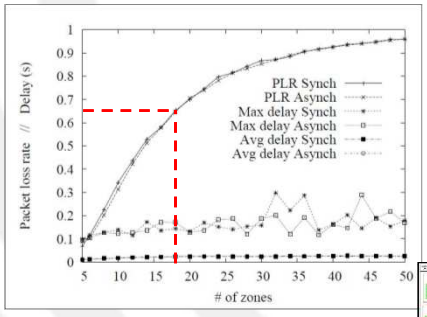

Unified Case study
reset_s0
wait_s1
receive s2
packet_rx:=packet_in
{intr_rx}
Abstraction
t4 RTL code
Emad Ebeid/ University of Verona
38/51

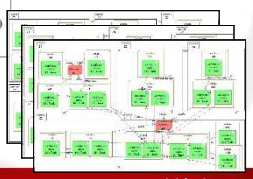

Unified Case study
«dataFlow»
«artifact»
«task»
t1
t2
t3
t4
t5
Behavioral Modeling
Functional simulation
«DataFlow»
requiredQoS = {(3,ms,(10,%),(20,KHz)}
task = {t3, t4}
«Task»
requiresMobility = false
CPU = 1
memorySize = (2,KB)
«Node»
task = []
networkInterface = [NWInterface1]
mobility = true
cost = (25,$)
gamma = [0.8,1.1]
memSize = (4,KB)
utilization = [(2)]
Emad Ebeid/ University of Verona
39/51


Unified Case study
Network synthesis
Network modeling
Network simulator (SCNSL)
Emad Ebeid/ University of Verona
40/51

Unified Case study

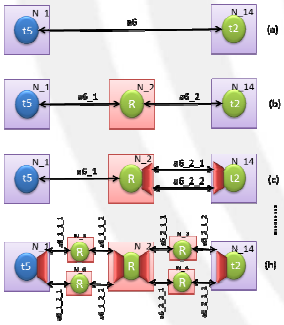
Network simulation statistics

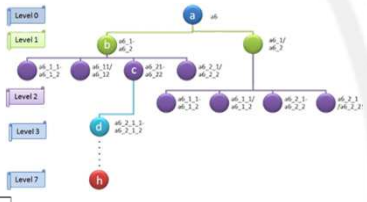


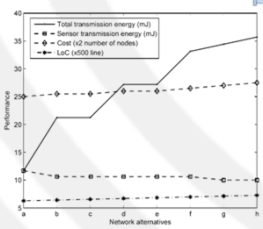


Unified Case study

Manipulation

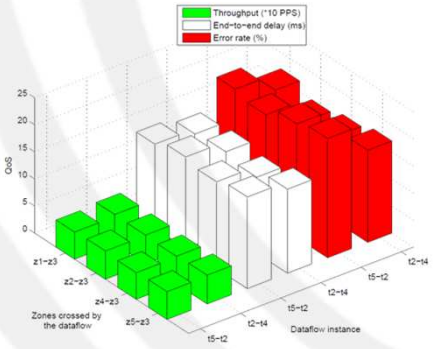






Unified Case study

Environmental effects



European projects involvement

- **COMPLEX** (COdesign and power Management in PLatform-based design space EXploration):
  - Network-aware Design-Space Exploration.
- **ToucHMore** (Automatic Customizable Tool-Chain for Heterogeneous Multicore Platform Software Development):
  - Abstraction of Heterogeneous IPs.
- ***SMAC*** *(SMArt Components & smart systems integration)*:
  - Environmental modeling.
- **CONTREX** (*COdesign and power Management in PLatform-based design space Exploration*):
  - Manipulation and a Framework for Design Space Exploration.

Reviewers remarks

- As Chapter 3 is one of the central chapters, I would appreciate to see a more general detailed overview/introduction on the individual mixed HW/SW design phases and an introduction the related terminology, i.e., *modeling, exploration, synthesis, validation.*
- A paragraph has been added to Section 3.1 to introduce the proposed methodology design phases.

Reviewers remarks

- As UML and its profiles cover a large set of different diagrams, a clear identification of the different types of *UML/SysML/MARTE diagrams* which are applied in each of the individual steps would be more appreciated.
- A paragraph has been added to Section 3.1.2 to highlight the used UML/SysML/MARTE diagrams.

Reviewers remarks

- In Section 1.3, the precision of objectives 1 and 2 may quite well benefit of some more *scientifically based justification*, consider i.e. to argue the impact of standards on industrial exploitation domains or benefits of model-based approached to support the proposed methodology.
- A paragraph has been added in sec 1.3, before the list of objectives, to clarify this point.

Publications & Submissions

- 11 conference papers
- 1 journal paper
- 2 submitted journal papers
- 1 submitted conference paper
- 2 PhD forums

Future research opportunities

- New manipulation rules for both structural and behavioral aspects of DES's
- Meta-models and design flow for mixed-criticalities DES
- Link between manipulation rules and optimization algorithms
- Extension to continuous time models by using SysML and SystemC AMS → dealing with Cyber-Physical Systems

Conclusions

- Methodology for DES's
- UML/*Network profile*
- Tool-chain for DSE and code generation
  - UML2HIF, NSM, HIF2UML and N2S
- Integration between DES's Simulators
  - SCNSL & SCoPE+
- Reused pre-designed and pre-verified IPs by abstracting and integrating them at UML level
- The overall design flow has been applied to a unified case study to evaluate its effectiveness and applicability


ESD
Thank you
for
your attention
Emad Ebeid/ University of Verona
51/51