UNIVERSITY OF TWENTE.

Freek van den Berg, Anne Remke and Boudewijn Haverkort
University of Twente, Centre for Telematics and Information Technology

iDSL: from performance measurements to predictions

Research question
What is the effect of merging frontal and lateral Image Processing onto a single PC for iXR machines?

Measurements

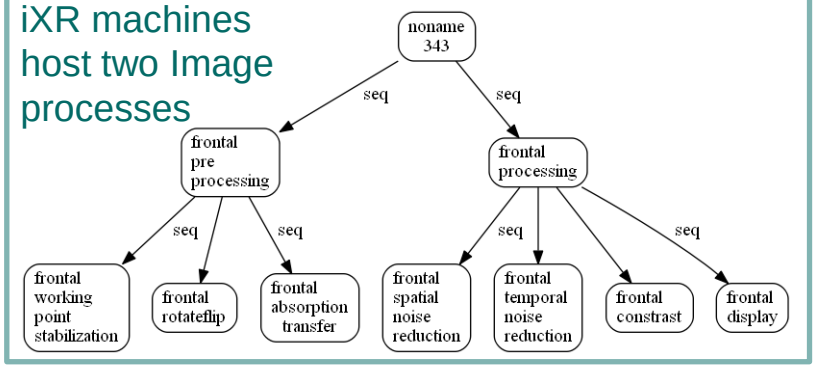
We calibrate our model with performance measurements of iXR systems

calibration

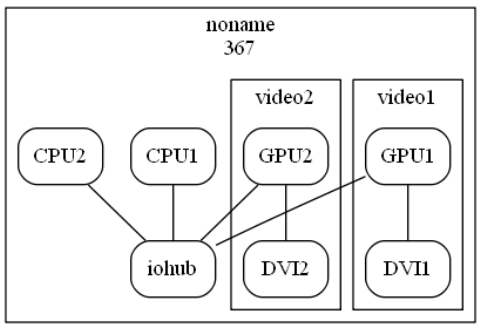
calibration

iDSL Process model

iXR machines host two Image processes



iDSL Resource model

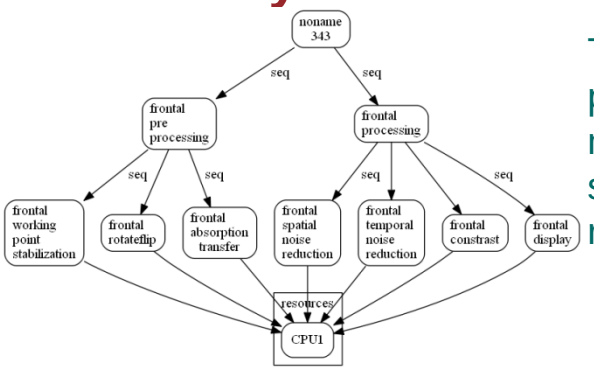


iXR machines are equiped with 2 CPUs, 2 GPUs and I/O hardware.

mapping

mapping

iDSL System model



The image processes map to the shared resources

instantiation

Tooling
iDSL, our performance evaluation language and toolbox, fully automates our approach

iDSL performance analysis

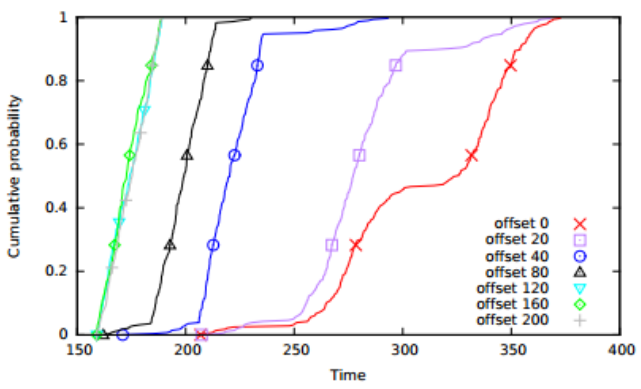
		Image frame rate				
		6	10	15	20	25
image resolution	512*512					
	1024*1024					
	2048*2048					

Performance analysis is performed for all 15 design instances

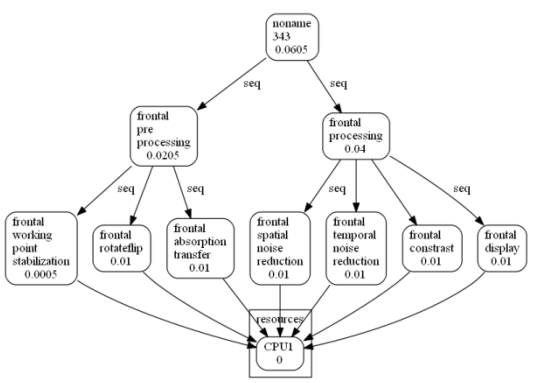
simulation, model checking

Results

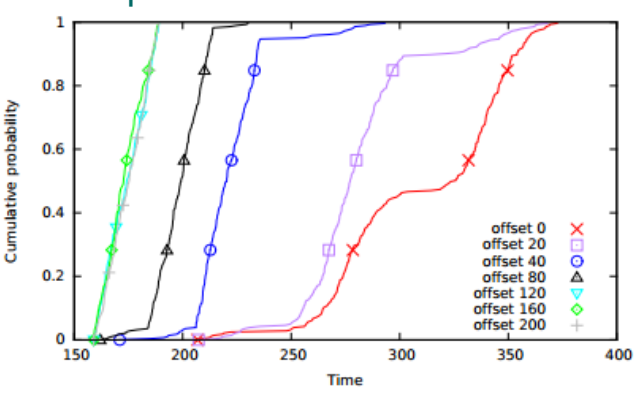
Comparing design alternatives conveys the impact of decisions



The latency breakdown displays the distribution of latency



Measurements and predictions are compared for model validation



Embedded Systems Innovation BY TNO

COMMIT/ A public- private research community

f.g.b.vandenberg@utwente.nl

UNIVERSITY OF TWENTE.

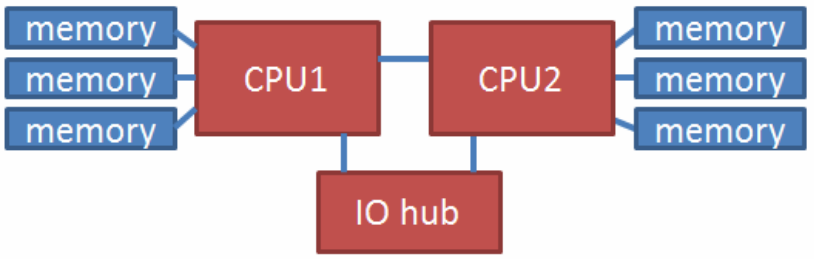
Freek van den Berg, Anne Remke and Boudewijn Haverkort
University of Twente, Centre for Telematics and Information Technology

iDSL: from performance measurements to predictions

Requirements

Characteristics of Allura Biplane machines:

- Functions execute in **constant time**
- Memory usage is **underspecified**
- Computations run in **parallel** on multiple CPUs
- Two **concurrent** applications share resources



Application

We analyze the performance of Allura Biplane machines:

What is the effect of **merging** frontal and lateral Image Processing onto a **single machine**?

Tool selection

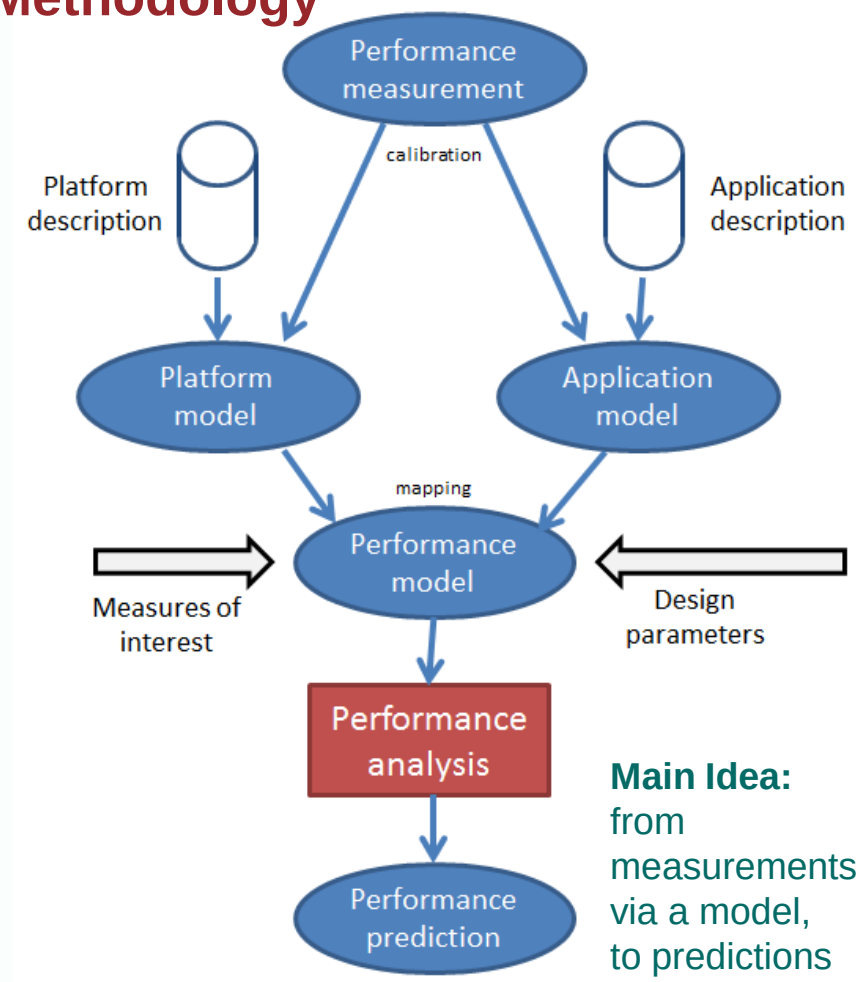
We use the MODEST toolset. It includes:

- **MODES** for simulations
- **UPPAAL** for model checking

The MODEST language supports:

- **Fixed delays**: for constant execution times
- **Non-deterministic choice**: for concurrency and underspecification
- **Parallelism**: for parallel computations

Methodology



Main Idea: from measurements, via a model, to predictions

Conceptual model

Key concepts of our approach are:

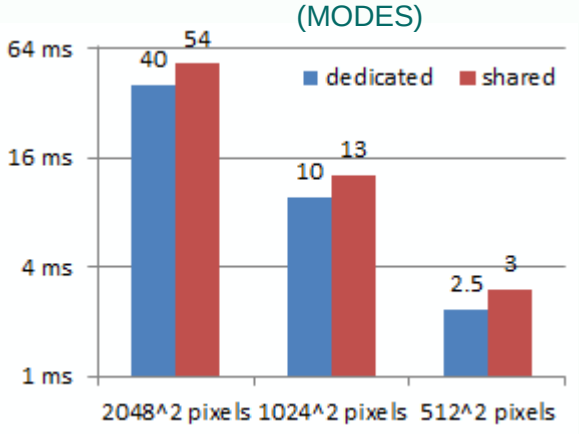
- **Application**: A hierarchical functional structure
- **Mapping**: Assigns an application to a platform
- **Scenario**: Execution of application instances, each based on a mapping.
- **Study**: Performs scenarios systematically to discover underlying system characteristics

Results

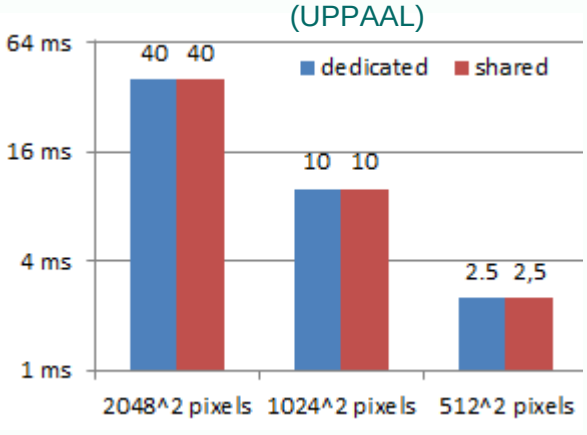
We show the **performance** of frontal Image Processing. We compare the dedicated and shared resource case, for three image resolutions.

Simulations yield **averages** (left), whereas model checking conveys **lower bounds** (right).

average processing times per image (MODES)



theoretical lower bounds per image (UPPAAL)



Embedded Systems Innovation BY TNO

COMMIT/ A public- private research community

f.g.b.vandenberg@utwente.nl

UNIVERSITY OF TWENTE.

Freek van den Berg, Anne Remke and Boudewijn Haverkort
University of Twente, Centre for Telematics and Information Technology

Hardware sharing demo: three experiments

Introduction

We present a MODEST performance model for Allura Biplane machines, which we demonstrate using three experiments.

For all the experiments we assume a 1024^2 image resolution at 15 FPS, Cardio2d, Cine Coronary Aorta and XRES3. Both frontal and lateral IP use these settings.

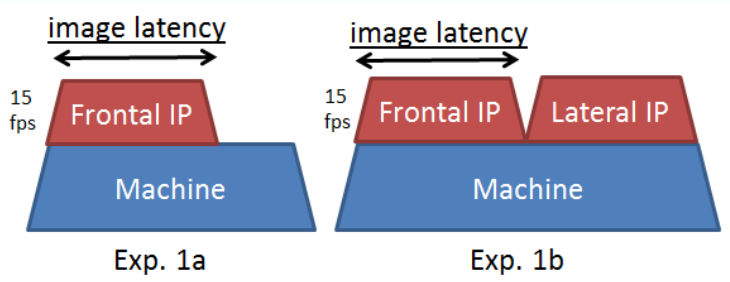
The three experiments address the image latency, throughput and resource utilization, obtained by means of simulation. Please feel free to write down the experiment results in the designated boxes.

Experiment 1: Image latency

The image latency (or delay) of Image Processing (IP) is the elapsed time between an image entering and leaving the system. We determine the effect of merging frontal and lateral IP onto a single machine.

In experiment **1a**, we run frontal IP dedicated on the machine, whereas in experiment **1b** both frontal and lateral IP are run on the machine.

By comparing the image latencies of both experiments, the effect of sharing hardware is determined.



1a ☐ 1b ☐

Conclusion

The MODEST performance model can be used to derive a variety of metrics, within a matter of seconds.

The experiments illustrate that the newly obtained performance insights can be used to aid decision making.

Experiment 2: Throughput

The throughput (or bandwidth) of Image Processing (IP) is the number of images the system can maximally process per second; We provide the system with a non-stop load.

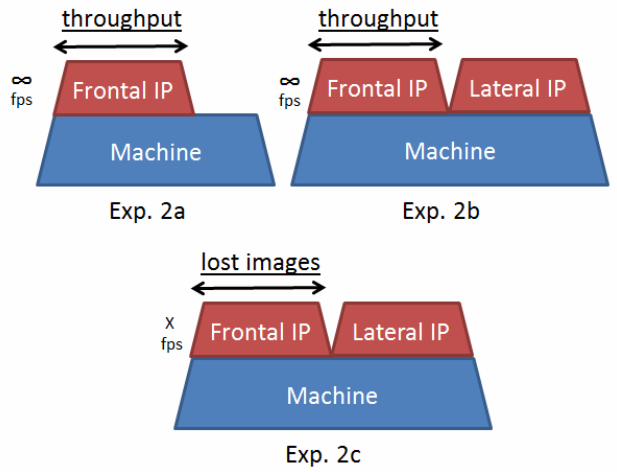
In experiment **2a**, we run frontal IP dedicated on the machine, whereas in experiment **2b** both frontal and lateral IP are run on the machine. Comparing both throughput values conveys the effect of hardware sharing.

In experiment **2c**, we test the outcome of experiment 2b by selecting a frame rate both just below and one just above the throughput. We count the number of lost images for each case.

2a ☐

2b ☐

2c ☐



Experiment 3: Resource utilization

The resource utilization of Image Processing (IP) is the relative time each resource is active, determined by monitoring each resource for (in)activity.

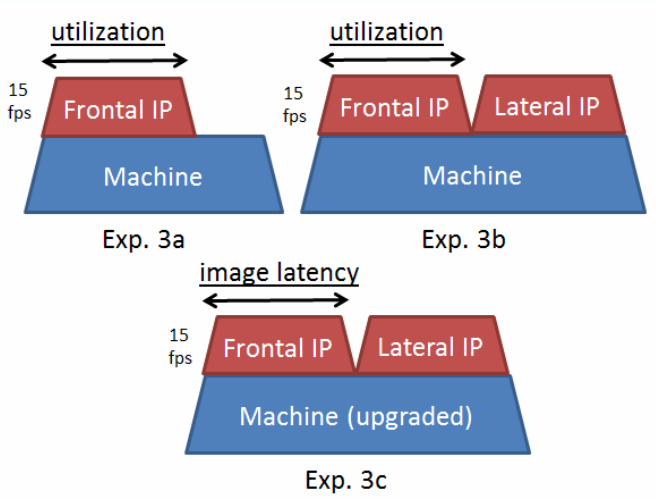
In experiment **3a**, we run frontal IP dedicated on the machine, whereas in experiment **3b** both frontal and lateral IP run on the machine.

We identify the bottleneck, the resource with the highest utilization. In experiment **3c**, we make this resource more powerful and assess the effect on latency.

3a ☐

3b ☐

3c ☐



Embedded Systems Innovation BY TNO

COMMIT/ A public- private research community

f.g.b.vandenberg@utwente.nl