



Universiteit
Leiden
The Netherlands

Improved hard real-time scheduling and transformations for embedded Streaming Applications

Spasic, J.

Citation

Spasic, J. (2017, November 14). *Improved hard real-time scheduling and transformations for embedded Streaming Applications*. Retrieved from <https://hdl.handle.net/1887/59459>

Version: Not Applicable (or Unknown)

License: [Licence agreement concerning inclusion of doctoral thesis in the Institutional Repository of the University of Leiden](#)

Downloaded from: <https://hdl.handle.net/1887/59459>

Note: To cite this publication please use the final published version (if applicable).

Cover Page



Universiteit Leiden



The following handle holds various files of this Leiden University dissertation:

<http://hdl.handle.net/1887/59459>

Author: Spasic, J.

Title: Improved hard real-time scheduling and transformations for embedded Streaming Applications

Issue Date: 2017-11-14

**Improved Hard Real-Time
Scheduling and Transformations
for Embedded Streaming Applications**

Jelena Spasić

Improved Hard Real-Time Scheduling and Transformations for Embedded Streaming Applications

PROEFSCHRIFT

ter verkrijging van
de graad van Doctor aan de Universiteit Leiden,
op gezag van Rector Magnificus Prof.mr. C.J.J.M. Stolker,
volgens besluit van het College voor Promoties
te verdedigen op dinsdag 14 november 2017
klokke 13:45 uur

door

Jelena Spasić
geboren te Trgovište, Servië
in 1984

Promotor:	Prof. dr. Joost N. Kok	Universiteit Leiden
Co-Promotor:	Dr. Todor P. Stefanov	Universiteit Leiden
Promotion Committee:	Prof. dr. Alix Munier Kordon	Université de Paris - LIP6
	Prof. dr. Petru Eles	Linköpings Universitet
	Dr. Andy Pimentel	Universiteit van Amsterdam
	Prof. dr. Aske Plaat	Universiteit Leiden
	Prof. dr. Jaap van den Herik	Universiteit Leiden
	Prof. dr. Harry Wijshoff	Universiteit Leiden

Improved Hard Real-Time Scheduling and Transformations
for Embedded Streaming Applications
Jelena Spasić. -
Dissertation Universiteit Leiden. - With ref. - With summary in Dutch.

Copyright © 2017 by Jelena Spasić. All rights reserved.

Cover designed by Miloš Ačanski.

This dissertation was typeset using L^AT_EX.

ISBN 978-94-6299-783-7

Printed by Ridderprint, Ridderkerk, The Netherlands.

Mojoj porodici
To my family

Contents

Table of Contents	vii
List of Figures	xi
List of Tables	xiii
List of Abbreviations	xv
1 Introduction	1
1.1 Trends in the Design of Embedded Streaming Systems	3
1.1.1 Platform Trend: Multi-Processor System-on-Chip (MPSoC)	3
1.1.2 Design Trend: Model-based Design Methodology	5
1.2 Design Requirements and Basic Approaches to Meet the Re- quirements	7
1.2.1 Timing Requirements	7
1.2.2 Energy Requirements	9
1.3 Problem Statement	10
1.3.1 Problem 1	10
1.3.2 Problem 2	10
1.3.3 Problem 3	11
1.3.4 Problem 4	12
1.4 Research Contributions	13
1.5 Thesis Outline	15
2 Background	17
2.1 Dataflow Models-of-Computations (MoCs)	17
2.1.1 Cyclo-Static Dataflow (CSDF)	18
2.1.2 Polyhedral Process Network (PPN)	20
2.2 Real-Time Scheduling Theory	22
2.2.1 Task Model	22

2.2.2	System Model	23
2.2.3	Real-Time Scheduling Algorithms	23
2.2.4	Uniprocessor Schedulability Analysis	24
2.2.5	Multiprocessor Schedulability Analysis	27
3	Hard Real-Time Scheduling Framework	31
3.1	Problem Statement	32
3.2	Contributions	32
3.3	Related Work	33
3.4	Motivational Example	35
3.5	Improved Hard Real-Time Scheduling of CSDF	37
3.5.1	Deriving Periods of Tasks	37
3.5.2	Deriving the Earliest Start Time of Actor's First Phase	41
3.5.3	Deriving Channel Buffer Sizes	44
3.5.4	Hard Real-Time Schedulability	46
3.5.5	Performance Analysis	47
3.5.6	Deriving the Number of Processors	52
3.6	Evaluation	54
3.6.1	Performance of the ISPS Approach	55
3.6.2	Time Complexity of the ISPS Approach	59
3.6.3	Reducing Latency under ISPS	61
3.7	Discussion	63
4	Exploiting Parallelism in Hard Real-Time Systems to Maximize Performance	65
4.1	Problem Statement	66
4.2	Contributions	67
4.3	Related Work	67
4.4	Motivational Example	69
4.5	New Unfolding Transformation for SDF Graphs	72
4.6	The Algorithm for Finding Proper Unfolding Factors	75
4.7	Evaluation	78
4.7.1	Efficiency of the Proposed Unfolding Transformation	79
4.7.2	Performance of Algorithm 4	80
4.7.3	Time Complexity of Algorithm 4	82
4.8	Discussion	82
5	Exploiting Parallelism in Hard Real-Time Systems to Minimize Energy	85
5.1	Problem Statement	86

5.2	Contributions	87
5.3	Related Work	87
5.4	Motivational Example	90
5.5	System Model	93
5.6	Energy Model	94
5.7	The Proposed Energy Minimization Approach	95
5.7.1	The Data-Parallel Energy Minimization Algorithm	95
5.7.2	Task Classification for Energy Minimization	98
5.7.3	Task Mapping for Energy Minimization	99
5.8	Evaluation	103
5.8.1	Comparison with [CKR14], [LSCS15], [SDK13] on Heterogeneous MPSoCs	104
5.8.2	Comparison with [Lee09] on Heterogeneous MPSoCs	106
5.8.3	Comparison on Homogeneous MPSoC	107
5.8.4	Overhead and Time Complexity Analysis	109
5.9	Discussion	110
6	An Accurate Energy Modeling of Streaming Systems	111
6.1	Problem Statement	111
6.2	Contributions	112
6.3	Related Work	113
6.4	System Model	115
6.4.1	Application Model	115
6.4.2	Platform Model	116
6.4.3	Application-to-Platform Mapping	117
6.5	Energy Model	118
6.5.1	Model Formulation	118
6.5.2	Derivation of Model Parameters	121
6.6	Evaluation of the Energy Model	127
6.7	Discussion	130
7	Summary and Conclusions	133
	Bibliography	137
	Samenvatting	153
	List of Publications	156
	Curriculum Vitae	159

Acknowledgments

161

List of Figures

1.1	An MPSoC platform example.	4
1.2	Motion JPEG encoder application.	6
2.1	A CSDF graph G	19
2.2	Example of a PPN (a) and the structure of process $P3$ (b).	21
3.1	(a) The SPS and (b) ISPS of graph G in Figure 2.1.	36
3.2	The periodic schedule σ for the CSDF graph G shown in Figure 2.1.	41
3.3	Production and consumption curves on edge $e_u = (v_i, v_j)$	50
4.1	An SDF graph G	69
4.2	Equivalent graphs of the SDF graph in Figure 4.1 by unfolding actor v_2 by factor 2 and v_3 by factor 3.	70
4.3	Unfolding channel e_2 from the graph in Figure 4.1 by using Algorithm 3 when $\vec{f} = [1, 2, 3, 1, 1]$	74
4.4	Comparison of our unfolding transformation to the approaches in [KM08], [FKBS11], [SLA12], [ZBS13].	80
4.5	Results of performance evaluation of our proposed approach in comparison to the approach in [ZBS13].	81
4.6	Results of time evaluation of our proposed approach in comparison to the approach in [ZBS13]	83
5.1	An SDF graph G	91
5.2	A CSDF graph G' obtained by unfolding SDF graph G in Figure 5.1 with $\vec{f} = [1, 2, 2, 1]$	91
5.3	Comparison of our proposed DPEM approach with related approaches on heterogeneous MPSoCs.	105
5.4	Comparison between DPEM and WYL on heterogeneous MP-SoCs.	107
5.5	Comparison on homogeneous MPSoC.	108

6.1 The read primitive implemented in software (a) and hardware
(b). 115

6.2 The architecture template of MPSoC platforms. 117

List of Tables

2.1	Summary of mathematical notations	17
3.1	Throughput, latency and number of processors for G under different scheduling schemes.	36
3.2	Benchmarks used for evaluation.	54
3.3	Comparison of different scheduling approaches.	58
3.4	Time complexity (in seconds) of different scheduling approaches.	58
3.5	Time complexity (in seconds) for the calculation of number of processors.	58
3.6	Performance of the ISPS approach under different latency constraints.	63
4.1	Results for G transformed by different transformation approaches.	71
4.2	Results for G transformed and mapped on 2 processors by different approaches.	71
4.3	Benchmarks used for evaluation.	79
5.1	Different MPSoC designs for G in Figure 5.1.	91
5.2	Benchmarks used for evaluation.	103
6.1	Accuracy of the energy model for CB, ShB and P2P MPSoC platforms	129
6.2	Accuracy of the energy estimation when contention is not considered in the model	130

List of Abbreviations

ADF	Affine Dataflow
BF	Best-Fit
BFD	Best-Fit Decreasing
CDP	Constrained-Deadline Periodic
CPU	Central Processing Unit
CSDF	Cyclo-Static Dataflow
DCT	Discrete Cosine Transform
DLP	Data-Level Parallelism
DM	Deadline Monotonic
DPEM	Data Parallel Energy Minimization
DSE	Design Space Exploration
EDF	Earliest Deadline First
EE	Energy Efficient
ESL	Electronic System-Level
FF	First-Fit
FFD	First-Fit Decreasing
FFID	First-Fit Increasing Deadlines
FIFO	First-In First-Out

FPGA	Field-Programmable Gate Array
GPU	Graphics Processing Unit
HSDF	Homogeneous SDF
ICP	Integer Convex Programming
IDP	Implicit-Deadline Periodic
ILP	Integer Linear Programming
ISA	Instruction-Set Architecture
ISPS	Improved Strictly Periodic Scheduling
ISS	Instruction Set Simulators
ITRS	International Technology Roadmap for Semiconductors
KPN	Kahn Process Network
LLF	Least Laxity First
LP	Linear Programming
LTE	Long-Term Evolution
MIDCP	Mixed Integer Disciplined Convex Programming
MJPEG	Motion JPEG
MoC	Model of Computation
MPSoC	Multi-Processor System-on-Chip
NoC	Network-on-Chip
NP	Non-deterministic Polynomial-time
PE	Performance Efficient
PLP	Pipeline-Level Parallelism
PM	Power Management
PPN	Polyhedral Process Network

PS	Periodic Scheduling
RM	Rate Monotonic
RSD	Reed Solomon Decoder
RTA	Response Time Analysis
RTL	Register-Transfer-Level
SDF	Synchronous Data Flow
SPS	Strictly Periodic Scheduling
STS	Self-timed Scheduling
TLP	Task-Level Parallelism
VFS	Voltage-Frequency Scaling
VLE	Variable Length Encoder
WCET	Worst-Case Execution Time
WF	Worst-Fit
WFD	Worst-Fit Decreasing

