



Universiteit  
Leiden  
The Netherlands

## Estimation and Optimization of the Performance of Polyhedral Process Networks

Haastregt, S. van

### Citation

Haastregt, S. van. (2013, December 17). *Estimation and Optimization of the Performance of Polyhedral Process Networks*. Retrieved from <https://hdl.handle.net/1887/22911>

Version: Corrected Publisher's Version

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

Downloaded from: <https://hdl.handle.net/1887/22911>

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

Cover Page



Universiteit Leiden



The handle <http://hdl.handle.net/1887/22911> holds various files of this Leiden University dissertation.

**Author:** Haastregt, Sven Joseph Johannes van

**Title:** Estimation and optimization of the performance of polyhedral process networks

**Issue Date:** 2013-12-17

# **Estimation and Optimization of the Performance of Polyhedral Process Networks**

Sven van Haastregt



# Estimation and Optimization of the Performance of Polyhedral Process Networks

## **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 17 december 2013  
klokke 12:30 uur

door

Sven van Haastregt  
geboren te Rijpwetering  
in 1985

## Samenstelling promotiecommissie:

promotor            Prof.dr. Ed Deprettere  
co-promotor        Dr. Bart Kienhuis

overige leden:    Prof.dr. Joost Kok  
                         Prof.dr. Harry Wijshoff  
                         Prof.dr. Koen Bertels      Technische Universiteit Delft  
                         Dr. Hristo Nikolov

This manuscript was edited by the author using Vim, and typeset using  $\LaTeX 2_{\epsilon}$ , BibTeX, and *MakeIndex* in a process automated using GNU Make. Graphics were produced mostly using Inkscape, and occasionally using Xfig or gnuplot. Git over ssh was used for revision tracking, synchronization, and backup purposes.

Cover design by Marcel IJssennagger.

Estimation and Optimization of the Performance of Polyhedral Process Networks  
Sven van Haastregt. -  
Thesis Universiteit Leiden. - With index, ref. - With summary in Dutch  
190 pages, 47988 words, 176 index entries, 162 references.  
ISBN 978-94-6182-383-0

Copyright ©2013 by Sven van Haastregt, Leiden, The Netherlands.

All rights reserved. No part of the material protected by this copyright notice may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without permission from the author.

Printed in the Netherlands.







# CONTENTS

<b>Contents</b>	<b>vii</b>
<b>Notation</b>	<b>xi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Problem Context . . . . .	1
1.2 Problem Statement . . . . .	4
1.3 Related Work . . . . .	7
1.3.1 High-Level Synthesis . . . . .	8
1.3.2 Electronic System-Level Synthesis . . . . .	10
1.4 Contributions and Outline . . . . .	12
<b>2 Background</b>	<b>15</b>
2.1 Polyhedral Model . . . . .	15
2.2 Models of Computation . . . . .	21
2.2.1 Homogeneous Synchronous Dataflow . . . . .	22
2.2.2 Synchronous Dataflow . . . . .	23
2.2.3 Cyclo-Static Dataflow . . . . .	25
2.2.4 Polyhedral Process Networks . . . . .	27
2.3 Derivation of PPNs from Sequential Programs . . . . .	30
2.3.1 Channel Type Determination . . . . .	31
2.3.2 Buffer Size Computation . . . . .	32
2.4 Code Generation . . . . .	33
2.4.1 Integrating Dedicated IP Cores . . . . .	34
<b>3 Synthesizing PPNs</b>	<b>37</b>
3.1 Motivation & Contributions . . . . .	37
3.2 IP Core Characterization . . . . .	38
3.2.1 IP Core Integration . . . . .	39

---

3.3	Data Reuse . . . . .	40
3.4	Sticky FIFOs . . . . .	42
3.5	Evaluation Logic Optimizations . . . . .	43
3.5.1	Pipelined Evaluation Logic . . . . .	44
3.5.2	ROM-Based Evaluation Logic . . . . .	44
3.5.3	Related Work . . . . .	47
3.6	Out-of-Order Communication . . . . .	48
3.7	Conclusion and Summary . . . . .	52
<b>4</b>	<b>Performance Estimation</b> . . . . .	<b>53</b>
4.1	Motivation . . . . .	53
4.2	Definitions . . . . .	55
4.3	RTL Simulation . . . . .	57
4.4	SystemC Simulation . . . . .	57
4.4.1	Cycle-Accurate Timed SystemC Simulation . . . . .	58
4.4.2	Light-weight Timed SystemC Simulation . . . . .	60
4.5	Maximum Cycle Mean Analysis . . . . .	61
4.5.1	Related Work . . . . .	61
4.5.2	Maximum Cycle Mean Analysis . . . . .	63
4.5.3	Derivation of PPN Modeling Graphs . . . . .	64
4.5.4	Case Studies . . . . .	72
4.6	Sequential Code Profiling . . . . .	75
4.6.1	Related Work . . . . .	76
4.6.2	Sequential Code Instrumentation of Static Programs . . . . .	77
4.6.3	Maximum Degree of Parallelism . . . . .	84
4.6.4	Absolute Throughput Estimation . . . . .	85
4.6.5	Case Study . . . . .	85
4.6.6	Transformation Performance Estimation . . . . .	90
4.6.7	Instrumentation Overhead . . . . .	92
4.7	Comparison . . . . .	94
4.8	Experimental Results . . . . .	96
4.8.1	Accuracy . . . . .	97
4.8.2	Running Time . . . . .	98
4.9	Conclusion and Summary . . . . .	99
<b>5</b>	<b>Application Transformation</b> . . . . .	<b>101</b>
5.1	Transformations . . . . .	101
5.1.1	Splitting . . . . .	101
5.1.2	Merging . . . . .	105

---

5.1.3	Stream Multiplexing . . . . .	106
5.1.4	Scheduling . . . . .	109
5.2	Transformation Efficacy Analysis . . . . .	116
5.2.1	Splitting . . . . .	116
5.2.2	Merging . . . . .	120
5.2.3	Stream Multiplexing . . . . .	121
5.2.4	Scheduling . . . . .	124
5.3	Conclusion and Summary . . . . .	127
<b>6</b>	<b>Industrial Case Study</b>	<b>129</b>
6.1	Sphere Decoding . . . . .	129
6.2	Reference Implementation . . . . .	131
6.3	AutoESL . . . . .	132
6.3.1	Design Flow . . . . .	133
6.3.2	Design Entry . . . . .	134
6.3.3	Design Productivity . . . . .	138
6.4	Daedalus . . . . .	140
6.4.1	Design Entry . . . . .	141
6.4.2	Synthesis . . . . .	147
6.5	Comparison . . . . .	148
6.6	Conclusion and Summary . . . . .	149
<b>7</b>	<b>Conclusions</b>	<b>151</b>
	<b>Samenvatting</b>	<b>155</b>
	<b>Curriculum Vitae</b>	<b>157</b>
	<b>Acknowledgments</b>	<b>159</b>
	<b>Bibliography</b>	<b>161</b>
	<b>Index</b>	<b>175</b>



# NOTATION

$ \cdot $	Cardinality: $ \mathcal{S}  \equiv$ the number of elements in $\mathcal{S}$ , page 20.
$[\cdot, \cdot)$	Interval: $[a, b) = \{x \in \mathbb{Z} \mid a \leq x < b\}$ .
$\lceil \cdot \rceil$	Least integer: $\lceil x \rceil = n \Leftrightarrow n \in \mathbb{N} \wedge n - 1 < x \leq n$ .
$\cdot \prec \cdot$	Lexicographical order, page 18.
$D_p$	Iteration domain of process $p$ , page 27.
$d(e)$	Number of initial tokens on edge $e$ , page 22.
$\delta_c$	Process reading from channel $c$ , page 27.
$\mathcal{E}$	The set of channels of a PPN, page 27.
$IIF$	Initiation interval of function $F$ , page 38.
$IPD_i^k$	$k$ -th Input Port Domain of process $i$ , page 27.
$\Lambda_F$	Latency (input-to-output delay) of function $F$ , page 38.
$M_c$	Channel relation of channel $c$ , page 27.
$\mathbb{N}$	The set of natural numbers, including 0.
$\mathbb{N}^+$	The set of positive natural numbers, excluding 0.
$OPD_i^k$	$k$ -th Output Port Domain of process $i$ , page 27.
$\mathcal{P}$	The set of processes of a PPN, page 27.
$\mathbb{Q}$	The set of rational numbers.
$\sigma_c$	Process writing to channel $c$ , page 27.
$T_p$	Period of a process $p$ , page 55.
$t(a)$	Execution time of data flow node $a$ , page 22.
$\tau_p$	Throughput of a process $p$ , page 55.
$\theta(\mathbf{i})$	Application of schedule $\theta$ to iteration vector $\mathbf{i}$ , pages 111, 112.
$\mathbb{Z}$	The set of integers.

