



Universiteit  
Leiden  
The Netherlands

## Parallelizing dynamic sequential programs using polyhedral process networks

Nadezhkin, D.

### Citation

Nadezhkin, D. (2012, December 20). *Parallelizing dynamic sequential programs using polyhedral process networks*. Retrieved from <https://hdl.handle.net/1887/20357>

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/1887/20357>

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

Cover Page



Universiteit Leiden



The handle <http://hdl.handle.net/1887/20357> holds various files of this Leiden University dissertation.

**Author:** Nadezhkin, Dmitry

**Title:** Parallelizing dynamic sequential programs using polyhedral process networks

**Issue Date:** 2012-12-20

# **Parallelizing Dynamic Sequential Programs using Polyhedral Process Networks**

Dmitry Nadezhkin



# Parallelizing Dynamic Sequential Programs using Polyhedral Process Networks.

## PROEFSCHRIFT

ter verkrijging van  
de graad van Doctor aan de Universiteit Leiden,  
op gezag van de Rector Magnificus prof. mr. P.F. van der Heijden,  
volgens besluit van het College voor Promoties  
te verdedigen op donderdag 20 december 2012  
klokke 10:00 uur

door

Dmitry Nadezhkin  
geboren te Arzamas-16, USSR  
in 1981

Samenstelling promotiecommissie:

|                |                              |                            |
|----------------|------------------------------|----------------------------|
| promotor       | Prof.dr.ir. Ed F. Deprettere | Universiteit Leiden        |
| co-promotor    | Dr.ir. Todor Stefanov        | Universiteit Leiden        |
| overige leden: | Prof.dr. Albert Cohen        | INRIA, Paris, France       |
|                | Prof.dr. Marco Bekooij       | Universiteit van Twente    |
|                | Dr. Andy D. Pimentel         | Universiteit van Amsterdam |
|                | Prof.dr. Joost Kok           | Universiteit Leiden        |
|                | Prof.dr. Harry Wijshoff      | Universiteit Leiden        |

Parallelizing Dynamic Sequential Programs using Polyhedral Process Networks  
Dmitry Nadezhkin. -  
Thesis Universiteit Leiden. - With ref. - With summary in Dutch  
ISBN 978-90-9027264-1

Copyright © 2012 by Dmitry Nadezhkin, 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, in-  
cluding photocopying, recording or by any information storage and retrieval system,  
without permission from the author.

Printed in the Netherlands

*To my parents Maria Alexandrovna and Alexander Trofimovich;  
and my Masha*



# Contents

|          |                                                                                                                      |           |
|----------|----------------------------------------------------------------------------------------------------------------------|-----------|
| <b>1</b> | <b>Introduction</b>                                                                                                  | <b>1</b>  |
| 1.1      | Automatic Derivation of Polyhedral Process Networks . . . . .                                                        | 3         |
| 1.1.1    | Dependence Analysis . . . . .                                                                                        | 4         |
| 1.1.2    | Linearization . . . . .                                                                                              | 7         |
| 1.2      | Problem statement . . . . .                                                                                          | 9         |
| 1.3      | Motivation and challenges . . . . .                                                                                  | 11        |
| 1.4      | Research Contributions . . . . .                                                                                     | 14        |
| 1.5      | Related Work . . . . .                                                                                               | 14        |
| 1.6      | Dissertation Outline . . . . .                                                                                       | 17        |
| <b>2</b> | <b>Background</b>                                                                                                    | <b>19</b> |
| 2.1      | Preliminaries . . . . .                                                                                              | 20        |
| 2.1.1    | Polyhedral Model . . . . .                                                                                           | 20        |
| 2.2      | The Program Model . . . . .                                                                                          | 22        |
| 2.3      | Polyhedral Process Networks . . . . .                                                                                | 24        |
| 2.4      | Exact Array Dataflow Analysis . . . . .                                                                              | 26        |
| 2.5      | Fuzzy Array Dataflow Analysis . . . . .                                                                              | 28        |
| 2.6      | Communication model identification in PPNs derived from static programs . . . . .                                    | 32        |
| <b>3</b> | <b>Automated Generation of Polyhedral Process Networks from Affine Nested-Loop Programs with Dynamic Loop Bounds</b> | <b>35</b> |

|          |                                                                                                                  |           |
|----------|------------------------------------------------------------------------------------------------------------------|-----------|
| 3.1      | Solution Overview . . . . .                                                                                      | 36        |
| 3.2      | Step 1 (Dynloop-to-WDP) . . . . .                                                                                | 37        |
| 3.3      | Step 2 (FADA analysis) . . . . .                                                                                 | 38        |
| 3.3.1    | Initial dSAC . . . . .                                                                                           | 39        |
| 3.4      | Step 3 (Control arrays) . . . . .                                                                                | 40        |
| 3.4.1    | Local control arrays . . . . .                                                                                   | 41        |
| 3.4.2    | Global control arrays . . . . .                                                                                  | 42        |
| 3.5      | Step 4 (PPN generation) . . . . .                                                                                | 43        |
| 3.5.1    | Topology creation of a PPN (substep 1) . . . . .                                                                 | 44        |
| 3.5.2    | Internal code structure generation (substep 2) . . . . .                                                         | 44        |
| 3.5.3    | Linearization (substep 3) . . . . .                                                                              | 45        |
| 3.6      | Calculation of deadlock-free buffer sizes . . . . .                                                              | 46        |
| 3.7      | Overhead Analysis . . . . .                                                                                      | 49        |
| 3.8      | Discussion and Summary . . . . .                                                                                 | 52        |
| <b>4</b> | <b>Automated Generation of Polyhedral Process Networks from<br/>Affine Nested-Loop Programs with While-loops</b> | <b>55</b> |
| 4.1      | Motivating example . . . . .                                                                                     | 56        |
| 4.2      | Solution Overview . . . . .                                                                                      | 57        |
| 4.3      | Step 1 (FADA analysis) . . . . .                                                                                 | 59        |
| 4.4      | Step 2 (Initial dSAC) . . . . .                                                                                  | 60        |
| 4.5      | Step 3 (Control variables) . . . . .                                                                             | 62        |
| 4.5.1    | Additional control variables . . . . .                                                                           | 63        |
| 4.6      | Step 4 (PPN generation) . . . . .                                                                                | 64        |
| 4.6.1    | Substep 1: Topology creation of a PPN . . . . .                                                                  | 65        |
| 4.6.2    | Substep 2: Code generation . . . . .                                                                             | 66        |
| 4.6.3    | Substep 3: Linearization . . . . .                                                                               | 67        |
| 4.6.4    | Substep 4: Implementation of a while-loop's iterator $w$ . . . . .                                               | 68        |
| 4.7      | Discussion and Summary . . . . .                                                                                 | 69        |
| <b>5</b> | <b>Identifying Communication Models in Polyhedral Process Networks de-<br/>rived from Dynamic Programs</b>       | <b>71</b> |

| <b>Contents</b>                                 | <b>ix</b>  |
|-------------------------------------------------|------------|
| 5.1 Rationale . . . . .                         | 72         |
| 5.2 Solution Approach . . . . .                 | 73         |
| 5.2.1 Parameterized mapping functions . . . . . | 75         |
| 5.3 Discussion and Summary . . . . .            | 78         |
| <b>6 Experimental Studies</b>                   | <b>79</b>  |
| 6.1 Low Speed Obstacle Detection . . . . .      | 79         |
| 6.2 Discussion and Summary . . . . .            | 86         |
| <b>7 Summary and Conclusions</b>                | <b>87</b>  |
| <b>Bibliography</b>                             | <b>91</b>  |
| <b>Index</b>                                    | <b>97</b>  |
| <b>Acknowledgments</b>                          | <b>99</b>  |
| <b>Samenvatting</b>                             | <b>101</b> |
| <b>Curriculum Vitae</b>                         | <b>103</b> |
| <b>List of Publications</b>                     | <b>105</b> |

