Reachability in Higher-Order-Counters

Alexander Heußner
University of Bamberg

Alexander Kartzow
University of Leipzig

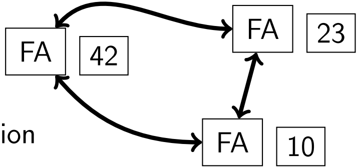
MFCS 2013

Table Of Contents

Table Of Contents

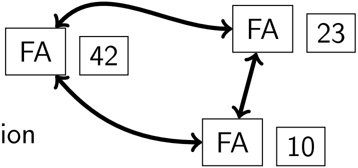
Alexander H.'s Point of Departure. . .

- distributed one counter systems
  that synchronize over counter values



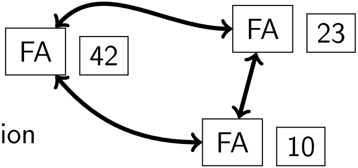
- safety verification & reachability question

Alexander H.'s Point of Departure...

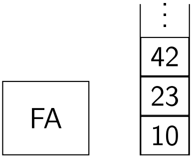


- distributed one counter systems
  that synchronize over counter values
- safety verification & reachability question
- idea: solve reachability by simulation via special register machines (that can "freeze" values for later resume)

Alexander H.'s Point of Departure...

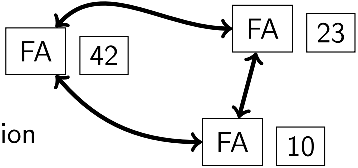


- distributed one counter systems
  that synchronize over counter values
- safety verification & reachability question
- idea: solve reachability by simulation via special register machines (that can "freeze" values for later resume)
- would need store hierarchy of registers, e.g., lifo:



Alexander H.'s Point of Departure. . .

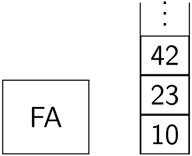
- distributed one counter systems
  that synchronize over counter values



- safety verification & reachability question

- idea: solve reachability by simulation via special register machines (that can "freeze" values for later resume)

- would need store hierarchy of registers, e.g., lifo:

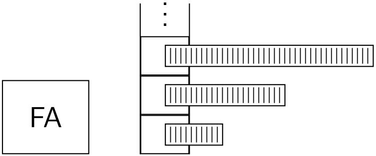


- what is known on these machines' safety decision problem?

…Joining the One of Alexander K.

- work on level k higher order pushdown automata (k-HOPA)

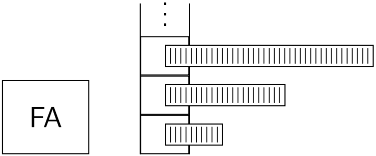
- 2-HOPA on unary alphabet lead to
  level 2 higher order **counter** automata (2-HOCA)



...Joining the One of Alexander K.

- work on level k higher order pushdown automata (k-HOPA)

- 2-HOPA on unary alphabet lead to
  level 2 higher order **counter** automata (2-HOCA)

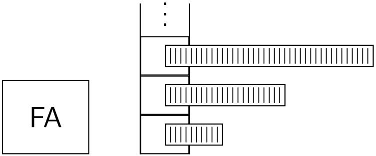


- why stop at level $k = 2$ here? ☺

. . . Joining the One of Alexander K.

- work on level k higher order pushdown automata (k-HOPA)

- 2-HOPA on unary alphabet lead to
  level 2 higher order **counter** automata (2-HOCA)

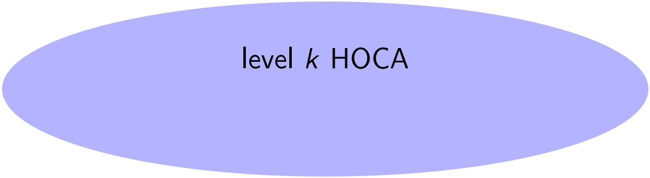


- why stop at level $k = 2$ here? ☺

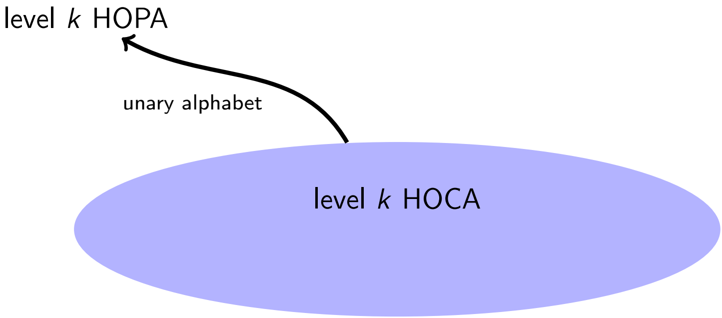
- what is known about level k higher order counter automata. . .
  - . . . regarding basic decision problems, e.g., (state) rechability?
  - . . . regarding their accepted languages?

Table Of Contents

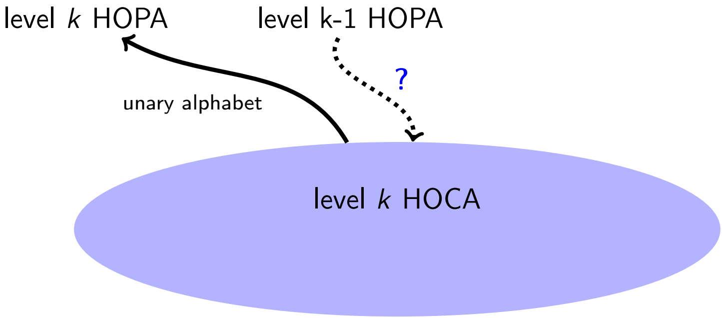
Higher-Order Counter Automata (HOCA)



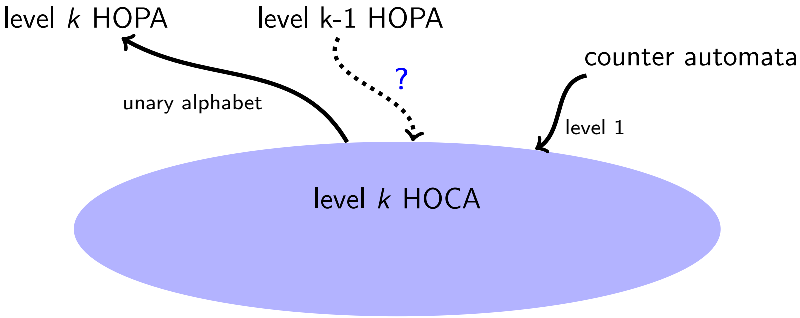
Higher-Order Counter Automata (HOCA)



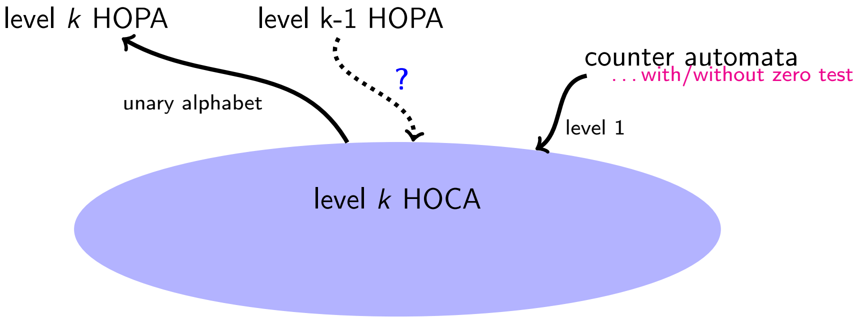
Higher-Order Counter Automata (HOCA)



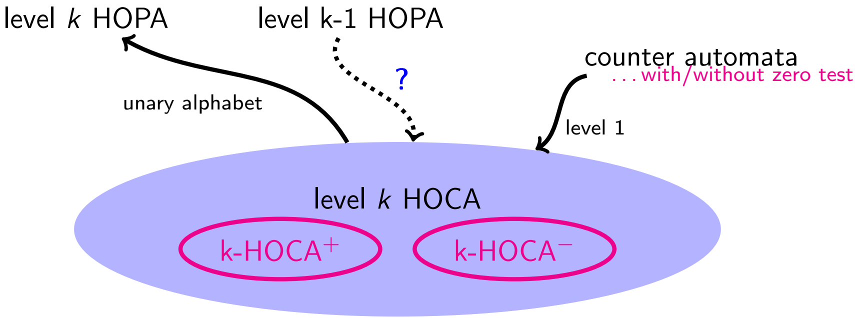
Higher-Order Counter Automata (HOCA)



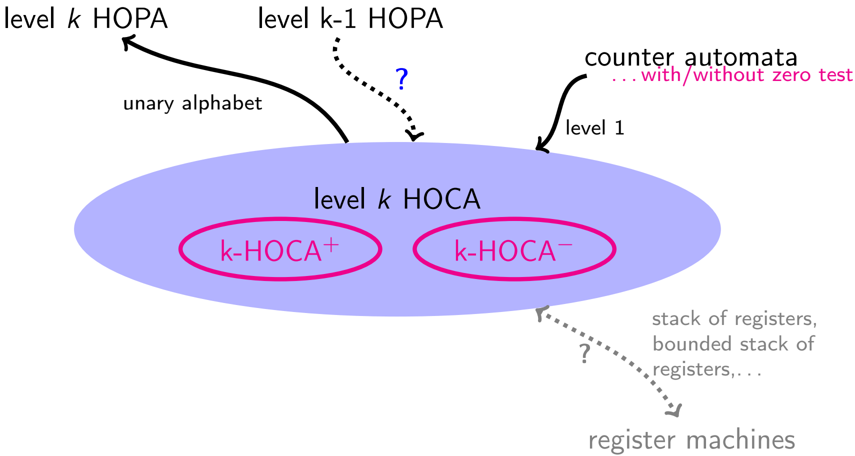
Higher-Order Counter Automata (HOCA)



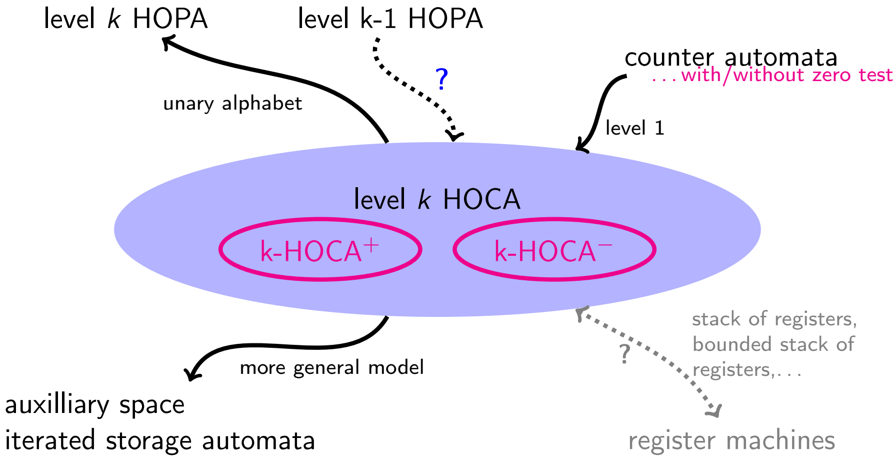
Higher-Order Counter Automata (HOCA)



Higher-Order Counter Automata (HOCA)



Higher-Order Counter Automata (HOCA)



Research Agenda

Pushdown vs. counter automata

- ➪ transfer ideas & results from pushdown to counter automata (e.g., similar algos often lead to better complexity results)
- ➪ lift results back to pushdown automata (e.g., for lower bounds)

Research Agenda

Pushdown vs. counter automata

- ➪ transfer ideas & results from pushdown to counter automata
  (e.g., similar algos often lead to better complexity results)
- ➪ lift results back to pushdown automata
  (e.g., for lower bounds)

Goal: higher-order pushdown vs. higher-order counter automata

1. adapt ideas/results from HOPA to HOCA
2. use newly derived results on HOCA to derive answers for important open questions on HOPA

Research Agenda

Pushdown vs. counter automata

- ➪ transfer ideas & results from pushdown to counter automata
  (e.g., similar algos often lead to better complexity results)
- ➪ lift results back to pushdown automata
  (e.g., for lower bounds)

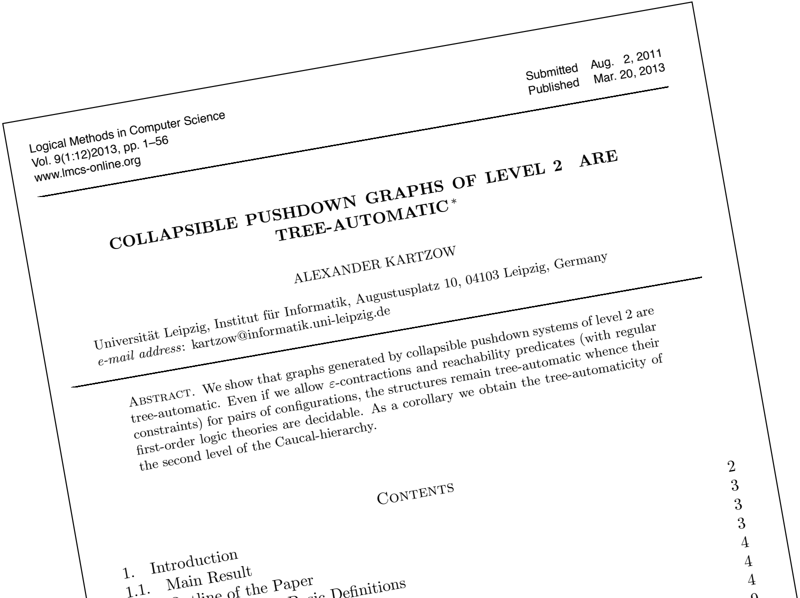
Goal: higher-order pushdown vs. higher-order counter automata

1. adapt ideas/results from HOPA to HOCA
2. use newly derived results on HOCA to derive answers for important open questions on HOPA

e.g., whether the hierarchy of collapsible pushdown languages is strict level-by-level

Table Of Contents

Theoretical Background



Logical Methods in Computer Science
Vol. 9(1:12)2013, pp. 1–56
www.lmcs-online.org

Submitted Aug. 2, 2011
Published Mar. 20, 2013

COLLAPSIBLE PUSHDOWN GRAPHS OF LEVEL 2 ARE TREE-AUTOMATIC*

ALEXANDER KARTZOW

Universität Leipzig, Institut für Informatik, Augustusplatz 10, 04103 Leipzig, Germany
e-mail address: kartzow@informatik.uni-leipzig.de

Abstract. We show that graphs generated by collapsible pushdown systems of level 2 are tree-automatic. Even if we allow ε-contractions and reachability predicates (with regular constraints) for pairs of configurations, the structures remain tree-automatic whence their first-order logic theories are decidable. As a corollary we obtain the tree-automaticity of the second level of the Caucal-hierarchy.

Contents

Regularity for level k HOPA

Theorem:

Let S be a collapsible pushdown system of level 2 with configuration graph G. Any expansion of the ϵ-contraction of G by regular reachability relations is tree-automatic.

Regularity for level k HOPA

Theorem:

Let S be a collapsible pushdown system of level 2 with configuration graph G. Any expansion of the ϵ-contraction of G by regular reachability relations is tree-automatic.

➪ return/loop construction

- finite memory cannot distinguish infinite number of stack configs
- runs of 2-HOPA from basic building blocks "returns" & "loops"
- reachability reduces to deciding whether certain returns/loops exist

Regularity for level k HOPA

Theorem:
Let S be a collapsible pushdown system of level 2 with configuration graph G. Any expansion of the ϵ-contraction of G by regular reachability relations is tree-automatic.

- ➪ return/loop construction
  - finite memory cannot distinguish infinite number of stack configs
  - runs of 2-HOPA from basic building blocks “returns” & “loops”
  - reachability reduces to deciding whether certain returns/loops exist
- ➪ apply binary tree-like encoding of level k stack

Theoretical Background (cont'd)

Logical Methods in Computer Science
Vol. 9(1:12)2013, pp. 1–56
www.lmcs-online.org

Submitted Aug. 2, 2011
Published Mar. 20, 2013

COLLAPSIBLE PUSHDOWN GRAPH
TREE-AUTOMAT

ALEXANDER KAR

Universität Leipzig, Institut für Informatik, Augustus
e-mail address: kartzow@informatik.uni-leipzig.de

Abstract. We show that graphs generated by
tree-automatic. Even if we allow ε-contractio
constraints) for pairs of configurations, the s
first-order logic theories are decidable. As
the second level of the Caucal-hierarchy.

INFORMATION AND COMPUTATION **95**, 21–75 (1991)

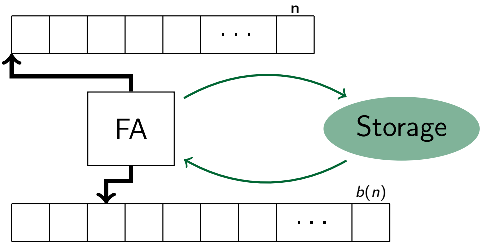
Iterated Stack Automata and Complexity Cl

Joost Engelfriet

Department of Computer Science, Leiden University,
P.O. Box 9512, 2300 RA Leiden, The Netherlands

An iterated pushdown is a pushdown of pushdowns of ... of pushdowns
iterated exponential function is 2 to the 2 to the ... to the 2 to some polyno
The main result presented here is that the nondeterministic 2-way and multi-
iterated pushdown automata characterize the deterministic iterated exponen
time complexity classes. This is proved by investigating both nondeterministic a
alternating auxiliary iterated pushdown automata, for which similar characteri
tion results are given. In particular it is shown that alternation corresponds to o
more iteration of pushdowns. These results are applied to the 1-way iterate
pushdown automata: (1) they form a proper hierarchy with respect to the numbe
of iterations, and (2) their emptiness problem is complete in deterministic
exponential time. Similar results are given for iterated st
erasing stack, nested stack, checking st
Press, Inc.

Auxiliary Space Storage Automata

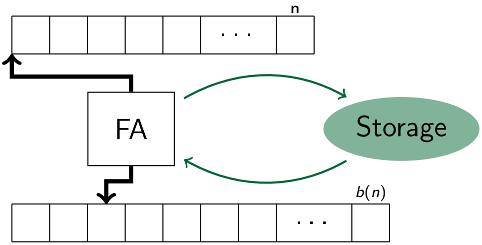


given by type definition
including tests and partial mappings of contents

Proposed general machine model:

- finite automata (can be determinstic, non-det., alternating)

Auxiliary Space Storage Automata

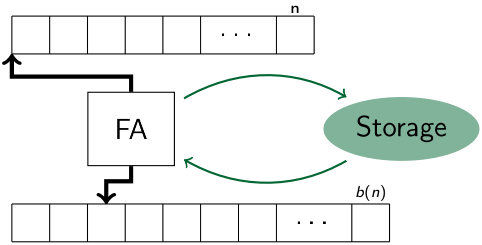


given by type definition
including tests and partial mappings of contents

Proposed general machine model:

- finite automata (can be determinstic, non-det., alternating)
- two-way read-only input tape of size n

Auxiliary Space Storage Automata

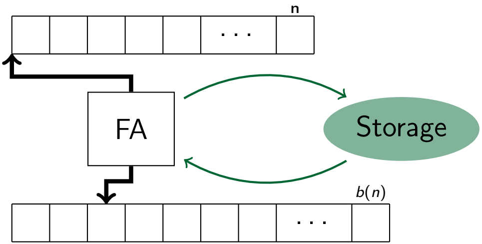


given by type definition
including tests and partial mappings of contents

Proposed general machine model:

- finite automata (can be determinstic, non-det., alternating)
- two-way read-only input tape of size n
- $SPACE(b(n))$ bounded worktape for given $b : \mathbb{N} \to \mathbb{N}$

Auxiliary Space Storage Automata



given by type definition
including tests and partial mappings of contents

Proposed general machine model:

- finite automata (can be determinstic, non-det., alternating)
- two-way read-only input tape of size n
- $SPACE(b(n))$ bounded worktape for given $b : \mathbb{N} \to \mathbb{N}$
- storage of a given (iterated) type,

HOCA as Iterated Stack Automata

Can render HOPA & HOCA as auxilliary space storage automata:

✓ finite control automaton

✓ one-way read-only input tape

↯ no working tape

✓ storage of following type:

- iteratively defined stack of … of stacks of a given basic type
- basic type is pushdown, or one counter with/without zero test

Engelfriet's Fundamental Insights

Main Result 1

Give alternative automata-theoretic characterization of determinstic exponential time complexity classes in terms of nondeterministic multi-head (k-1)-iterated pushdown automata. Give similar characterizations for non-determistic and alternation versions of the automata.

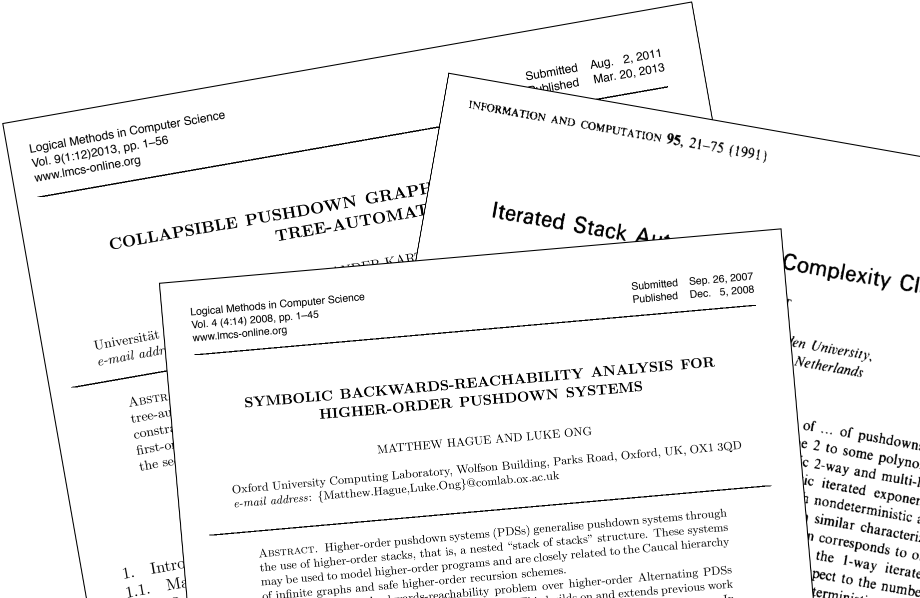
Main Result 2

Research the emptiness problem for iterated pushdown automata of various kinds.

From the (dense, technical & elegant) proofs...

- trade off: decrease nestedness of storage by one level requires exponentially more space
- trade alternation for non-determinism for another level of nestedness

Theoretical Background (cont'd)



Theoretical Background (cont'd)

Theoretical Background (cont'd)

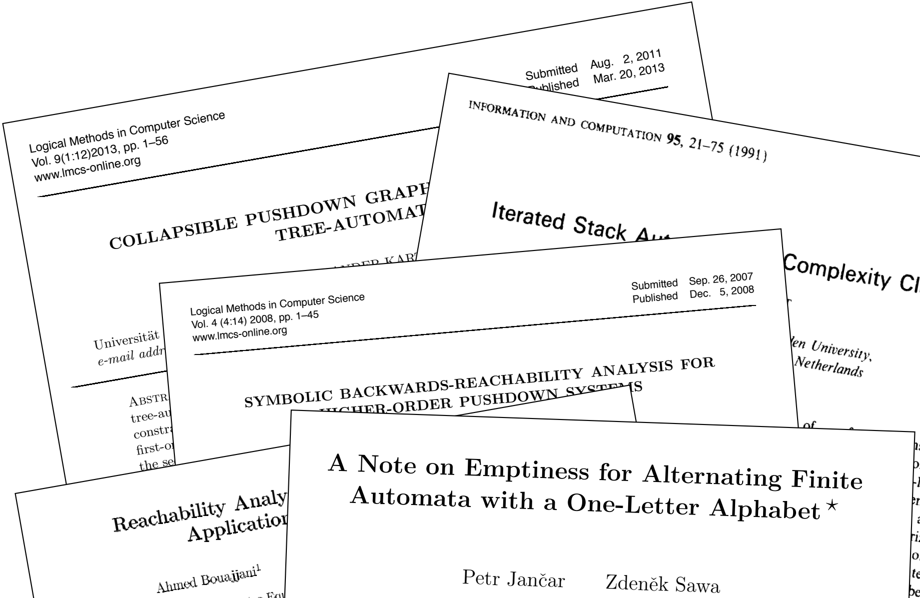


Table Of Contents

Level-2 HOCA State Reachability

Proposition:

Control state reachability for 2-HOCA$^-$ is complete for **P**.

proof idea:

- ➪ extend returns & loops construction leads to algorithm in **P**
- ➪ 2-HOCA$^-$ can simulate 1-HOPA ✓
- ➪ hardness from known results on PDA (i.e., 1-HOPA)

Level-2 HOCA Regular Reachability

Theorem:

Regular fwd/bwd reachability for 2-HOCA$^-$ is complete for **P**.

proof idea:

- turn reachability predicate on 2-HOCA$^-$ into tree automatic-relation
- can avoid exponential blow-up
- reduce to previous returns & loops construction for state reachability

Level-2 HOCA Regular Reachability

Theorem:

Regular fwd/bwd reachability for 2-HOCA$^-$ is complete for **P**.

proof idea:

- ➪ turn reachability predicate on 2-HOCA$^-$ into tree automatic-relation
- ➪ can avoid exponential blow-up
- ➪ reduce to previous returns & loops construction for state reachability

How to define regularity ?

- ➪ based on binary tree encoding of [Kartzow]
- ➪ substantially different from "weaker & more succinct" regularity...
  - ...via 2-store automata [Bouajjani/Meyer]
  - ...via sequence of pushdown operations [Carayol]

Level-k HOCA$^+$ State Reachability

Theorem:

Alternating control state reachability of k-HOCA with zero test is complete for **DSPACE**$(\bigcup_{d\in\mathbb{N}} \exp_{k-2}(n^d))$.

$exp_0(n) := n$

$exp_{k+1}(n) := exp(exp_k(n))$

proof idea:

- reduce to membership of alternating exp_{k-3} space storage automata
- lower bound by reduction similar to proof of [Jancar/Sawa] for **PSPACE**-C of non-emptiness of alternating automata

Level-k HOCA$^+$ State Reachability

Theorem:

Alternating control state reachability of k-HOCA with zero test is complete for **DSPACE**$(\bigcup_{d\in\mathbb{N}} \exp_{k-2}(n^d))$.

$exp_0(n) := n$

$exp_{k+1}(n) := exp(exp_k(n))$

proof idea:

- reduce to membership of alternating exp_{k-3} space storage automata
- lower bound by reduction similar to proof of [Jancar/Sawa] for **PSPACE**-C of non-emptiness of alternating automata

Theorem:

Control state reachability of k-HOCA with zero test is complete for **DSPACE**$(\bigcup_{d\in\mathbb{N}} \exp_{k-1}(n^d))$.

- lift reduction results of [Engelfriet] to our setting
- lower bound from (k-1)-HOPA

Level-k HOCA$^-$ State Reachability

Theorem:

Alternating control state reachability of k-HOCA without zero test is complete for **DTIME**$(\bigcup_{d \in \mathbb{N}} \exp_{k-2}(n^d))$.

proof idea:

- adapt [Engelfriet]'s inductive reductions with counter automata reachability as basic case
- hardness from simulating (k-1)-HOPA

Level-k HOCA^- State Reachability

Theorem:

Alternating control state reachability of k-HOCA without zero test is complete for **DTIME**$(\bigcup_{d\in\mathbb{N}} \mathsf{exp}_{k-2}(n^d))$.

proof idea:

- adapt [Engelfriet]'s inductive reductions with counter automata reachability as basic case
- hardness from simulating (k-1)-HOPA

Theorem:

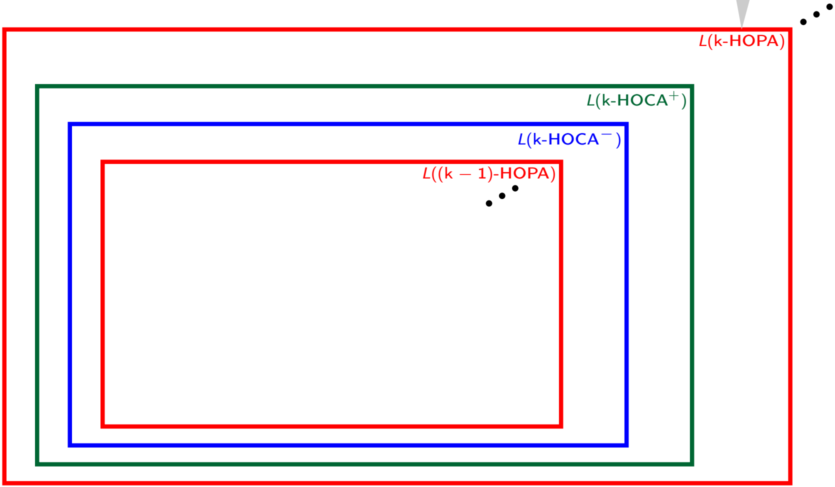
Control state reachability of k-HOCA without zero test is complete for **DTIME**$(\bigcup_{d\in\mathbb{N}} \mathsf{exp}_{k-1}(n^d))$.

- treat alternation for space

The Big Picture

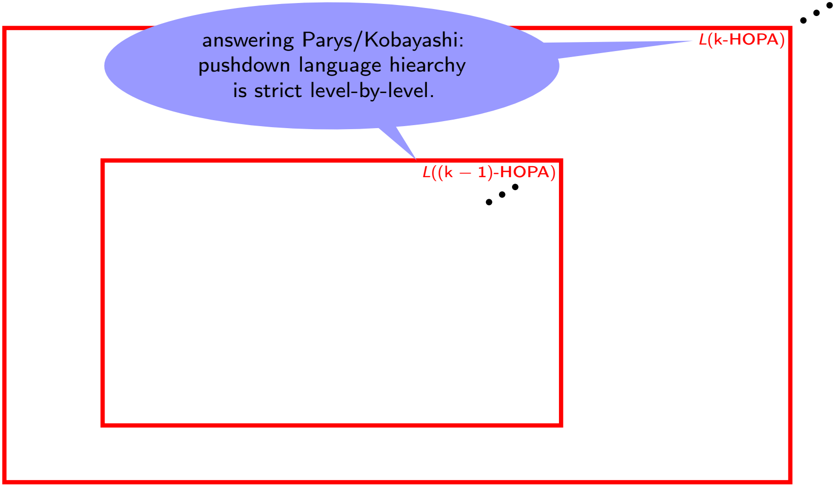
$L(-)$ stands for language accepted by class "$-$"

If **DTIME**$(\bigcup_{d\in\mathbb{N}} \exp_k(n^d)) \subsetneq$ **DSPACE**$(\bigcup_{d\in\mathbb{N}} \exp_k(n^d)) \subsetneq$ **DTIME**$(\bigcup_{d\in\mathbb{N}} \exp_{k+1}(n^d))$, then. . .



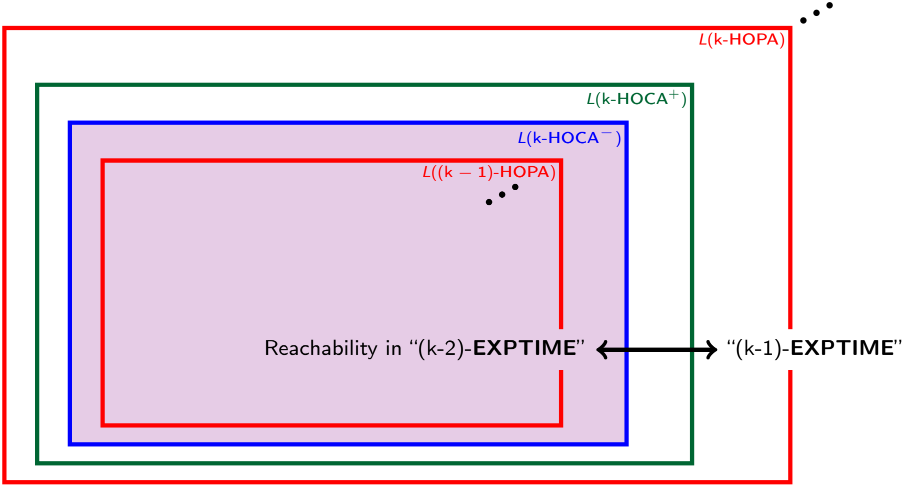
The Big Picture

If $\mathbf{DTIME}(\bigcup_{d\in\mathbb{N}} \exp_k(n^d)) \subsetneq \mathbf{DSPACE}(\bigcup_{d\in\mathbb{N}} \exp_k(n^d)) \subsetneq$ $\mathbf{DTIME}(\bigcup_{d\in\mathbb{N}} \exp_{k+1}(n^d))$, then...



The Big Picture

If $\mathbf{DTIME}(\bigcup_{d\in\mathbb{N}} \exp_k(n^d)) \subsetneq \mathbf{DSPACE}(\bigcup_{d\in\mathbb{N}} \exp_k(n^d)) \subsetneq \mathbf{DTIME}(\bigcup_{d\in\mathbb{N}} \exp_{k+1}(n^d))$, then...



More Directions on Level-2 HOCA

... let's get back to a finite set of lifo stored registers

- let c be a natural number (encoded in unary/binary)
- c-bounded reachability problem: can a given state be reached by a run where all level 2 stacks are at most of height c?
- bounded reachability problem: c part of input...

More Directions on Level-2 HOCA

…let's get back to a finite set of lifo stored registers

- let c be a natural number (encoded in unary/binary)
- c-bounded reachability problem: can a given state be reached by a run where all level 2 stacks are at most of height c?
- bounded reachability problem: c part of input…

Preliminary results:

	c-bounded	bounded (unary)	bounded (binary)
2-HOPA	**EXPTIME**-C		
2-HOCA$^+$	**PSPACE**-C	**PSPACE**-C	
2-HOCA$^-$	**NL**-C	**P**-C	**P**-C

Another Important Side Result

Insight

Digging in "old"[†] papers and adapting their ideas to current problems is inevitable and helps to avoid reinventing the wheel several times — also in computer science.

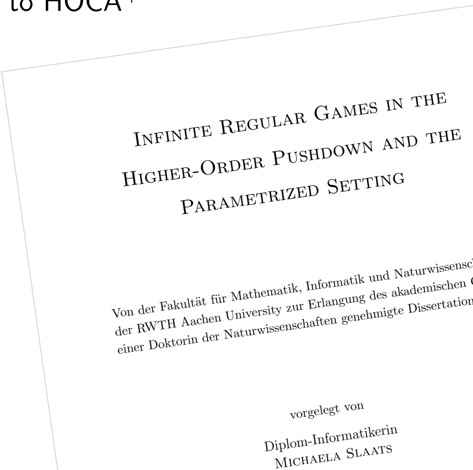
[†] in comuter science, papers older than 15–20 years are already seen as stone-age ☺

Table Of Contents

Related Work

PhD thesis of Michaela Slaats 2012

➪ higher order counters equivalent to HOCA$^+$


INFINITE REGULAR GAMES IN THE
HIGHER-ORDER PUSHDOWN AND THE
PARAMETRIZED SETTING

Von der Fakultät für Mathematik, Informatik und Naturwissensch
der RWTH Aachen University zur Erlangung des akademischen
einer Doktorin der Naturwissenschaften genehmigte Dissertation

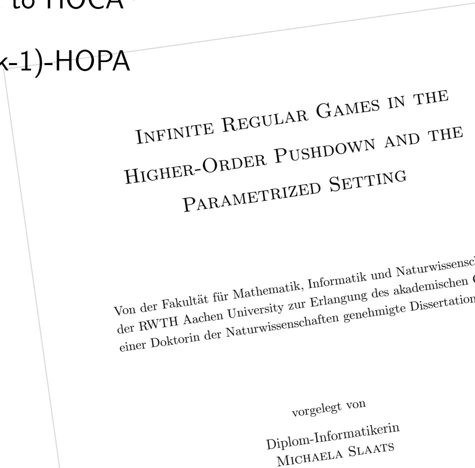
vorgelegt von

Diplom-Informatikerin
MICHAELA SLAATS

Related Work

PhD thesis of Michaela Slaats 2012

- ➪ higher order counters equivalent to HOCA$^+$
- ➪ show: k-HOCA$^+$ can simulate (k-1)-HOPA
- ➪ we shown:
  already holds for k-HOCA$^-$ ✓



INFINITE REGULAR GAMES IN THE HIGHER-ORDER PUSHDOWN AND THE PARAMETRIZED SETTING

Von der Fakultät für Mathematik, Informatik und Naturwissensc
der RWTH Aachen University zur Erlangung des akademischen
einer Doktorin der Naturwissenschaften genehmigte Dissertation

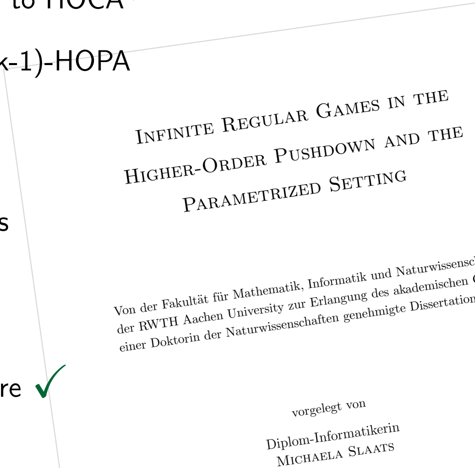
vorgelegt von

Diplom-Informatikerin
MICHAELA SLAATS

Related Work

PhD thesis of Michaela Slaats 2012

- ➪ higher order counters equivalent to HOCA$^+$
- ➪ show: k-HOCA$^+$ can simulate (k-1)-HOPA
- ➪ we shown:
  already holds for k-HOCA$^-$ $\checkmark$
- ➪ k-HOCA$^+$ as language acceptors
- ➪ conjecture:
  $L(\text{k-HOCA}^+) \subsetneq L(\text{k-HOPA})$
- ➪ we confirm and extend conjecture $\checkmark$


INFINITE REGULAR GAMES IN THE
HIGHER-ORDER PUSHDOWN AND THE
PARAMETRIZED SETTING

Von der Fakultät für Mathematik, Informatik und Naturwissensch
der RWTH Aachen University zur Erlangung des akademischen G
einer Doktorin der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Diplom-Informatikerin
MICHAELA SLAATS

Ongoing & Future Work

- does 2-HOCA$^-$ generalize the class of PDA while retaining their well-known good algorithmic properties?
- try to extend positive results for μ-calculus model checking from pushdown automata to 2-HOCA
- regular reachability question for other notions of regularity
- apply (bounded) 2-HOCA as register machines for the verification of concurrent systems
- ...