Demystifying Reachability in Vector Addition Systems

J. Leroux S. Schmitz

LaBRI, CNRS LSV, ENS Cachan & INRIA

LICS 2015 in Kyoto, July 6th, 2015

Outline

vector addition systems (VAS)
and their reachability problem

… solved by the KLMST algorithm
of Sacerdote and Tenney (1977), Mayr (1981), Kosaraju (1982), and Lambert (1992)

decomposition theorem
the KLMST algorithm constructs an ideal decomposition of the set of runs

upper bound theorem
VAS reachability is in cubic Ackermann

Outline

vector addition systems (VAS)
and their reachability problem

... solved by the KLMST algorithm
of Sacerdote and Tenney (1977), Mayr (1981), Kosaraju (1982), and Lambert (1992)

decomposition theorem
the KLMST algorithm constructs an ideal decomposition of the set of runs

upper bound theorem
VAS reachability is in cubic Ackermann

Outline

vector addition systems (VAS)
and their reachability problem

. . . solved by the KLMST algorithm
of Sacerdote and Tenney (1977), Mayr (1981), Kosaraju (1982), and Lambert (1992)

decomposition theorem
the KLMST algorithm constructs an ideal decomposition of the set of runs

upper bound theorem
VAS reachability is in cubic Ackermann

Vector Addition Systems (VAS)

(Karp and Miller, 1969)

Syntax

- dimension $\mathrm{d} \in \mathbb{N}$
- finite set $\mathbf{A} \subseteq_{\mathrm{fin}} \mathbb{Z}^{\mathrm{d}}$ of actions $\mathbf{a} \in \mathbf{A}$

Semantics

- configurations $\mathbf{u}, \mathbf{v}, \ldots \in \mathbb{N}^{\mathrm{d}}$
- transitions $\mathbf{u} \xrightarrow{\mathbf{a}} \mathbf{v} \in \mathbb{N}^{\mathrm{d}} \times \mathbf{A} \times \mathbb{N}^{\mathrm{d}}$ with $\mathbf{v} = \mathbf{u} + \mathbf{a}$

Vector Addition Systems (VAS)

(Karp and Miller, 1969)

Syntax

- dimension $d \in \mathbb{N}$
- finite set $\mathbf{A} \subseteq_{\text{fin}} \mathbb{Z}^d$ of actions $\mathbf{a} \in \mathbf{A}$

Semantics

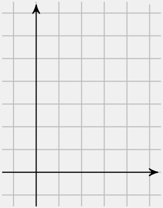
- configurations $\mathbf{u}, \mathbf{v}, \ldots \in \mathbb{N}^d$
- transitions $\mathbf{u} \xrightarrow{\mathbf{a}} \mathbf{v} \in \mathbb{N}^d \times \mathbf{A} \times \mathbb{N}^d$ with $\mathbf{v} = \mathbf{u} + \mathbf{a}$

Example VAS

Example

$d = 2$

$\mathbf{A} = \{$  ,  $\}$

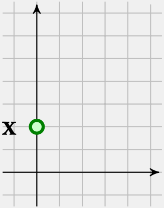


Example VAS

Example

$d = 2$

$\mathbf{A} = \{$  ,  $\}$



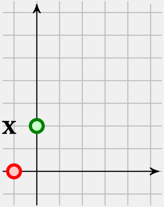
$\mathbf{x}=(0,2)$

Example VAS

Example

$d = 2$

$\mathbf{A} = \{$  ,  $\}$



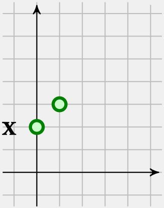
$\mathbf{x} = (0,2) \longrightarrow (-1,0) \notin \mathbb{N}^2$

Example VAS

Example

$d = 2$

$\mathbf{A} = \{$  ,  $\}$



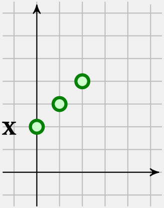
$\mathbf{x}=(0,2)$  $\longrightarrow (1,3)$

Example VAS

Example

$d = 2$

$\mathbf{A} = \Big\{$ ,  $\Big\}$



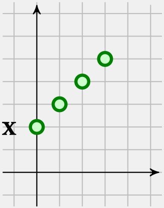
$\mathbf{x} = (0,2)$  $\longrightarrow (1,3)$  $\longrightarrow (2,4)$

Example VAS

Example

$d = 2$

$\mathbf{A} = \{$  ,  $\}$

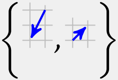


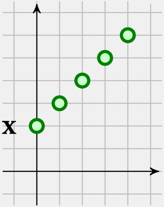
$\mathbf{x}=(0,2)$  $\longrightarrow (1,3)$  $\longrightarrow (2,4)$  $\longrightarrow (3,5)$

Example VAS

Example

$d = 2$

$\mathbf{A} =$ 

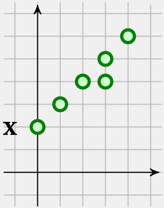


$\mathbf{x} = (0,2) \longrightarrow (1,3) \longrightarrow (2,4) \longrightarrow (3,5) \longrightarrow (4,6)$

Example VAS

Example

$d = 2$ $\mathbf{A} = \{$  ,  $\}$



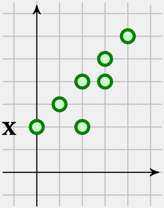
$\mathbf{x} = (0,2) \longrightarrow (1,3) \longrightarrow (2,4) \longrightarrow (3,5) \longrightarrow (4,6) \longrightarrow (3,4)$

Example VAS

Example

$d = 2$

$\mathbf{A} = \{$  $\}$

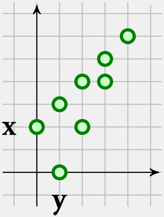


$\mathbf{x} = (0,2) \longrightarrow (1,3) \longrightarrow (2,4) \longrightarrow (3,5) \longrightarrow (4,6) \longrightarrow (3,4) \longrightarrow (2,2)$

Example VAS

Example

$d = 2$ $\mathbf{A} = \{ \swarrow, \nearrow \}$



$\mathbf{x} = (0,2) \xrightarrow{\nearrow} (1,3) \xrightarrow{\nearrow} (2,4) \xrightarrow{\nearrow} (3,5) \xrightarrow{\nearrow} (4,6) \xrightarrow{\swarrow} (3,4) \xrightarrow{\swarrow} (2,2) \xrightarrow{\swarrow} (0,1) = \mathbf{y}$

Runs and Preruns

Definition (Prerun)

A prerun is an element

$$(\mathbf{u},\ (\mathbf{u}_1,\mathbf{a}_1,\mathbf{v}_1)\cdots(\mathbf{u}_k,\mathbf{a}_k,\mathbf{v}_k),\ \mathbf{v})$$

from $\text{PreRuns}_{\mathbf{A}} \stackrel{\text{def}}{=} \mathbb{N}^d \times (\mathbb{N}^d \times \mathbf{A} \times \mathbb{N}^d)^* \times \mathbb{N}^d$

Definition (Run)

A prerun is connected (is a run) if

(source) $\mathbf{u} = \mathbf{u}_1$

(transitions) $\forall 1 \leqslant j \leqslant k, \mathbf{u}_j + \mathbf{a}_j = \mathbf{v}_j$

(contiguity) $\forall 1 < j \leqslant k, \mathbf{v}_{j-1} = \mathbf{u}_j$

(target) $\mathbf{v}_k = \mathbf{v}$

The Reachability Problem

$\text{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \stackrel{\text{def}}{=} \{\rho \in \text{PreRuns}_{\mathbf{A}} \mid \rho \text{ is a run with source } \mathbf{x} \text{ and target } \mathbf{y}\}$

VAS Reachability

input $\mathbf{A} \subseteq_{\text{fin}} \mathbb{Z}^{d}, \mathbf{x}, \mathbf{y} \in \mathbb{N}^{d}$

question Is $\mathbf{y}$ reachable from $\mathbf{x}$ in $\mathbf{A}$?
I.e., is $\text{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \neq \emptyset$?

Theorem (Mayr, 1981; Kosaraju, 1982; Lambert, 1992; Leroux, 2011)
VAS Reachability is decidable.

- by the KLMST algorithm (Mayr, 1981; Kosaraju, 1982; Lambert, 1992)
- by Presburger invariants (Leroux, 2011)

The Reachability Problem

$\text{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \stackrel{\text{def}}{=} \{\rho \in \text{PreRuns}_{\mathbf{A}} \mid \rho \text{ is a run with source } \mathbf{x} \text{ and target } \mathbf{y}\}$

VAS Reachability

input $\mathbf{A} \subseteq_{\text{fin}} \mathbb{Z}^d, \mathbf{x},\mathbf{y} \in \mathbb{N}^d$

question Is $\mathbf{y}$ reachable from $\mathbf{x}$ in $\mathbf{A}$?
I.e., is $\text{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \neq \emptyset$?

Theorem (Mayr, 1981; Kosaraju, 1982; Lambert, 1992; Leroux, 2011)

VAS Reachability is decidable.

- by the KLMST algorithm (Mayr, 1981; Kosaraju, 1982; Lambert, 1992)
- by Presburger invariants (Leroux, 2011)

The Reachability Problem

$\mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \stackrel{\text{def}}{=} \{\rho \in \mathrm{PreRuns}_{\mathbf{A}} \mid \rho \text{ is a run with source } \mathbf{x} \text{ and target } \mathbf{y}\}$

VAS Reachability

input $\mathbf{A} \subseteq_{\text{fin}} \mathbb{Z}^d, \mathbf{x},\mathbf{y} \in \mathbb{N}^d$

question Is $\mathbf{y}$ reachable from $\mathbf{x}$ in $\mathbf{A}$?
I.e., is $\mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \neq \emptyset$?

Theorem (Mayr, 1981; Kosaraju, 1982; Lambert, 1992; Leroux, 2011)

VAS Reachability is decidable.

- by the KLMST algorithm (Mayr, 1981; Kosaraju, 1982; Lambert, 1992)
- by Presburger invariants (Leroux, 2011)

The Reachability Problem

$\text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y}) \stackrel{\text{def}}{=} \{\rho \in \text{PreRuns}_{\mathbf{A}} \mid \rho \text{ is a run with source } \mathbf{x} \text{ and target } \mathbf{y}\}$

VAS Reachability

input $\mathbf{A} \subseteq_{\text{fin}} \mathbb{Z}^d, \mathbf{x}, \mathbf{y} \in \mathbb{N}^d$

question Is $\mathbf{y}$ reachable from $\mathbf{x}$ in $\mathbf{A}$?
I.e., is $\text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y}) \neq \emptyset$?

Theorem (Mayr, 1981; Kosaraju, 1982; Lambert, 1992; Leroux, 2011)

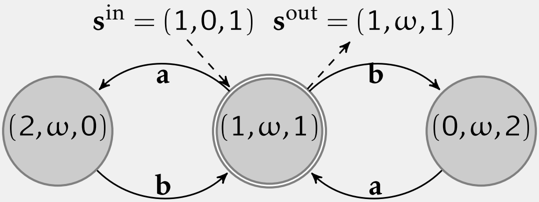
VAS Reachability is decidable.

- by the KLMST algorithm (Mayr, 1981; Kosaraju, 1982; Lambert, 1992)
- by Presburger invariants (Leroux, 2011)

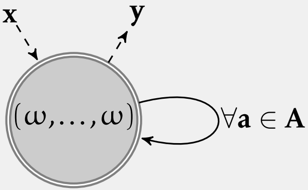
Marked Graph

Example

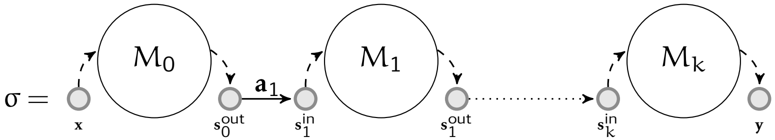
$$\mathbf{A} = \{\mathbf{a} = (1,1,-1), \mathbf{b} = (-1,0,1))$$



Example (Initial graph)

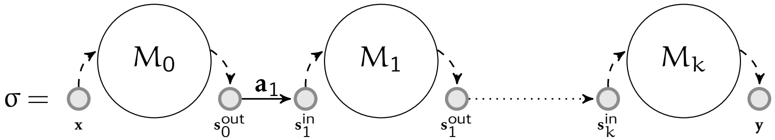


Marked Graph Sequence



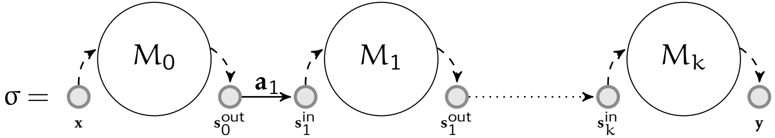
- associated set of runs $\Omega_\sigma \subseteq \mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y})$
- perfectness condition (aka θ condition): decidable semantic condition ensuring $\Omega_\sigma \neq \emptyset$
- effective decomposition of imperfect sequences: $\Omega_\sigma = \bigcup_{\sigma' \in \mathrm{decompose}(\sigma)} \Omega_{\sigma'}$

Marked Graph Sequence



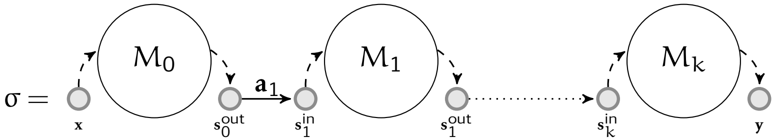
- associated set of runs $\Omega_\sigma \subseteq \mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y})$
- perfectness condition (aka θ condition): decidable semantic condition ensuring $\Omega_\sigma \neq \emptyset$
- effective decomposition of imperfect sequences: $\Omega_\sigma = \bigcup_{\sigma' \in \mathrm{decompose}(\sigma)} \Omega_{\sigma'}$

Marked Graph Sequence



- associated set of runs $\Omega_\sigma \subseteq \text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$
- perfectness condition (aka θ condition): decidable semantic condition ensuring $\Omega_\sigma \neq \emptyset$
- effective decomposition of imperfect sequences: $\Omega_\sigma = \bigcup_{\sigma' \in \text{decompose}(\sigma)} \Omega_{\sigma'}$

Marked Graph Sequence



- associated set of runs $\Omega_\sigma \subseteq \mathsf{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$
- perfectness condition (aka θ condition): decidable semantic condition ensuring $\Omega_\sigma \neq \emptyset$
- effective decomposition of imperfect sequences: $\Omega_\sigma = \bigcup_{\sigma' \in \mathrm{decompose}(\sigma)} \Omega_{\sigma'}$

KLMST Algorithm (Schematically)

Construct a sequence $S_0, S_1, \ldots$ of finite sets of marked witness graph sequences with $\forall n$

$$\Omega_n \stackrel{\text{def}}{=} \bigcup_{\sigma \in S_n} \Omega_\sigma = \text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$$

init S_0 is s.t. $\text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y}) = \Omega_0$

$\forall n$

- if $S_n = \{\sigma\} \uplus S$ and $\neg\text{perfect}(\sigma)$

  $$S_{n+1} \stackrel{\text{def}}{=} S \cup (\text{decompose}(\sigma))$$

- otherwise stop: $\text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y}) \neq \emptyset$

terminates via a ranking function r

$$\forall \sigma' \in \text{decompose}(\sigma) \,.\, r(\sigma) > r(\sigma')$$

KLMST Algorithm (Schematically)

Construct a sequence $S_0, S_1, \ldots$ of finite sets of marked witness graph sequences with $\forall n$

$$\Omega_n \stackrel{\text{def}}{=} \bigcup_{\sigma \in S_n} \Omega_\sigma = \text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$$

init S_0 is s.t. $\text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y}) = \Omega_0$

$\forall n$
- if $S_n = \{\sigma\} \uplus S$ and $\neg\text{perfect}(\sigma)$

$$S_{n+1} \stackrel{\text{def}}{=} S \cup (\text{decompose}(\sigma))$$

- otherwise stop: $\text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y}) \neq \emptyset$

terminates via a ranking function r

$$\forall \sigma' \in \text{decompose}(\sigma) \,.\, r(\sigma) > r(\sigma')$$

KLMST Algorithm (Schematically)

Construct a sequence $S_0, S_1, \ldots$ of finite sets of marked witness graph sequences with $\forall n$

$$\Omega_n \stackrel{\text{def}}{=} \bigcup_{\sigma \in S_n} \Omega_\sigma = \text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$$

init S_0 is s.t. $\text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y}) = \Omega_0$

$\forall n$
- if $S_n = \{\sigma\} \uplus S$ and $\neg\text{perfect}(\sigma)$

  $S_{n+1} \stackrel{\text{def}}{=} S \cup (\text{decompose}(\sigma))$
- otherwise stop: $\text{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y}) \neq \emptyset$

terminates via a ranking function r

$\forall \sigma' \in \text{decompose}(\sigma) \,.\, r(\sigma) > r(\sigma')$

KLMST Algorithm (Schematically)

Construct a sequence $S_0, S_1, \ldots$ of finite sets of marked witness graph sequences with $\forall n$

$$\Omega_n \stackrel{\text{def}}{=} \bigcup_{\sigma \in S_n} \Omega_\sigma = \mathrm{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$$

init S_0 is s.t. $\mathrm{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y}) = \Omega_0$

$\forall n$
- if $S_n = \{\sigma\} \uplus S$ and $\neg\mathrm{perfect}(\sigma)$

  $S_{n+1} \stackrel{\text{def}}{=} S \cup (\mathrm{decompose}(\sigma))$
- otherwise stop: $\mathrm{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y}) \neq \emptyset$

terminates via a ranking function r

$\forall \sigma' \in \mathrm{decompose}(\sigma) \,.\, r(\sigma) > r(\sigma')$

Mysteries

1. conceptual complexity

   the complexity of the proofs (especially of Mayr, 1981) wrap the result in mystery use of their original ideas has been made

2. computational complexity
   - ExpSpace-hard problem (Lipton, 1976),
   - Ackermann lower bound on the KLMST algorithm (Müller, 1985)
   - no known upper bound

MYSTERIES

1. conceptual complexity

   "the complexity of the two proofs (especially of Mayr, 1981) wrapped the result in mystery and no use of their original ideas has been made" *(Lambert, 1992)*

2. computational complexity
   - EXPSPACE-hard problem (Lipton, 1976),
   - ACKERMANN lower bound on the KLMST algorithm (Müller, 1985)
   - no known upper bound

Mysteries

1. conceptual complexity

   "the complexity of the two proofs (especially of Mayr, 1981) wrapped the result in mystery and no use of their original ideas has been made" *(Lambert, 1992)*

2. computational complexity
   - ExpSpace-hard problem (Lipton, 1976),
   - Ackermann lower bound on the KLMST algorithm (Müller, 1985)
   - no known upper bound

Mystery 1: Conceptual Complexity

Theorem (Decomposition Theorem)

The KLMST algorithm computes the ideal decomposition of

$$\downarrow \mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \stackrel{def}{=} \{\rho' \in \mathrm{PreRuns}_{\mathbf{A}} \mid \exists \rho \in \mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \,.\, \rho' \trianglelefteq \rho\}$$

Deciphering the statement (upcoming slides)

- definition of a well quasi order (wqo) over preruns (Jančar, 1990)
- wqo ideals (Finkel and Goubault-Larrecq, 2009, 2012)

Mystery 1: Conceptual Complexity

Theorem (Decomposition Theorem)

The KLMST algorithm computes the ideal decomposition of

$$\downarrow \mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \stackrel{def}{=} \{\rho' \in \mathrm{PreRuns}_{\mathbf{A}} \mid \exists \rho \in \mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \,.\, \rho' \trianglelefteq \rho\}$$

Deciphering the statement (upcoming slides)

- definition of a well quasi order (wqo) over preruns (Jančar, 1990)
- wqo ideals (Finkel and Goubault-Larrecq, 2009, 2012)

Mystery 1: Conceptual Complexity

Theorem (Decomposition Theorem)

The KLMST algorithm computes the ideal decomposition of

$$\downarrow \mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \stackrel{def}{=} \{\rho' \in \mathrm{PreRuns}_{\mathbf{A}} \mid \exists \rho \in \mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y})\,.\,\rho' \trianglelefteq \rho\}$$

Deciphering the statement (upcoming slides)

- definition of a well quasi order (wqo) over preruns (Jančar, 1990)
- wqo ideals (Finkel and Goubault-Larrecq, 2009, 2012)

Mystery 1: Conceptual Complexity

Theorem (Decomposition Theorem)

The KLMST algorithm computes the ideal decomposition of

$$\downarrow \mathsf{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \stackrel{def}{=} \{\rho' \in \mathsf{PreRuns}_{\mathbf{A}} \mid \exists \rho \in \mathsf{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) \,.\, \rho' \trianglelefteq \rho\}$$

Significance

- entails decidability of VAS Reachability:

$$\mathsf{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) = \emptyset \text{ iff } \downarrow \mathsf{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y}) = \emptyset$$

- generalises Habermehl et al. (2010)'s result on the computability of downward-closures of VAS languages
- template for decidability proofs in extensions (unordered data nets, branching VAS, pushdown VAS, ...)?

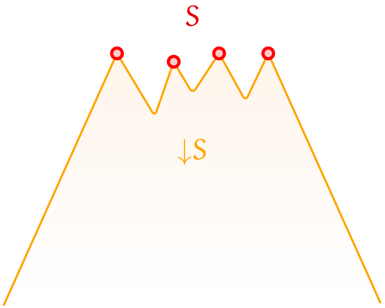
Well quasi orders

- Downward closure
  $\downarrow S \stackrel{\text{def}}{=} \{x \in X \mid \exists s \in S \,.\, x \leqslant s\}$

over a quasi-order $(X, \leqslant)$

- Descending Chain Property
  A quasi-order $(X, \leqslant)$ is a well quasi order if every descending chain $D_0 \supsetneq D_1 \supsetneq \cdots$ of downwards-closed subsets of X is finite.

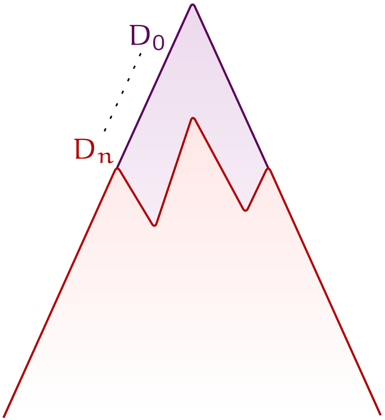
- Examples
  - finite sets with equality
  - $\mathbb{N}$ with the naturals ordering
  - $A \times B$ with the product ordering (Dickson's Lemma)
  - A^* with scattered subword ordering (Higman's Lemma)



Well quasi orders

- Downward closure
  $\downarrow S \stackrel{\text{def}}{=} \{x \in X \mid \exists s \in S \,.\, x \leqslant s\}$ over a quasi-order $(X, \leqslant)$

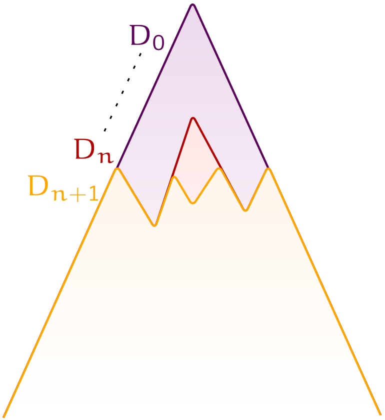
- Descending Chain Property
  A quasi-order $(X, \leqslant)$ is a well quasi order if every descending chain $D_0 \supsetneq D_1 \supsetneq \cdots$ of downwards-closed subsets of X is finite.



- Examples
  - finite sets with equality
  - $\mathbb{N}$ with the naturals ordering
  - $A \times B$ with the product ordering (Dickson's Lemma)
  - A^* with scattered subword ordering (Higman's Lemma)

Well quasi orders

- Downward closure
  $\downarrow S \stackrel{\text{def}}{=} \{x \in X \mid \exists s \in S \,.\, x \leqslant s\}$ over a quasi-order $(X, \leqslant)$
- Descending Chain Property
  A quasi-order $(X, \leqslant)$ is a well quasi order if every descending chain $D_0 \supsetneq D_1 \supsetneq \cdots$ of downwards-closed subsets of X is finite.
- Examples
  - finite sets with equality
  - $\mathbb{N}$ with the naturals ordering
  - $A \times B$ with the product ordering (Dickson's Lemma)
  - A^* with scattered subword ordering (Higman's Lemma)



Well quasi orders

- Downward closure
  $\downarrow S \stackrel{\text{def}}{=} \{x \in X \mid \exists s \in S \,.\, x \leqslant s\}$
- Descending Chain Property
  A quasi-order $(X, \leqslant)$ is a well quasi order if every descending chain $D_0 \supsetneq D_1 \supsetneq \cdots$ of downwards-closed subsets of X is finite.
- Examples
  - finite sets with equality
  - $\mathbb{N}$ with the naturals ordering
  - $A \times B$ with the product ordering (Dickson's Lemma)
  - A^* with scattered subword ordering (Higman's Lemma)

WELL QUASI ORDERS

- Downward closure
  $\downarrow S \stackrel{\text{def}}{=} \{x \in X \mid \exists s \in S \,.\, x \leqslant s\}$
- Descending Chain Property
  A quasi-order $(X, \leqslant)$ is a well quasi order if every descending chain $D_0 \supsetneq D_1 \supsetneq \cdots$ of downwards-closed subsets of X is finite.
- Examples
  - finite sets with equality
  - $\mathbb{N}$ with the naturals ordering
  - $A \times B$ with the product ordering (Dickson's Lemma)
  - A^* with scattered subword ordering (Higman's Lemma)

Well quasi orders

- Downward closure
  $\downarrow S \overset{\text{def}}{=} \{x \in X \mid \exists s \in S \,.\, x \leqslant s\}$
- Descending Chain Property
  A quasi-order $(X, \leqslant)$ is a well quasi order if every descending chain $D_0 \supsetneq D_1 \supsetneq \cdots$ of downwards-closed subsets of X is finite.
- Examples
  - finite sets with equality
  - $\mathbb{N}$ with the naturals ordering
  - $A \times B$ with the product ordering (Dickson's Lemma) (for A, B two wqos)
  - A^* with scattered subword ordering (Higman's Lemma)

Well quasi orders

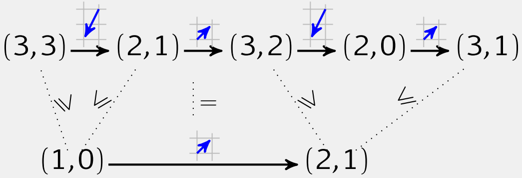
- Downward closure
  $\downarrow S \stackrel{\text{def}}{=} \{x \in X \mid \exists s \in S \,.\, x \leqslant s\}$
- Descending Chain Property
  A quasi-order $(X, \leqslant)$ is a well quasi order if every descending chain $D_0 \supsetneq D_1 \supsetneq \cdots$ of downwards-closed subsets of X is finite.
- Examples
  - finite sets with equality
  - $\mathbb{N}$ with the naturals ordering
  - $A \times B$ with the product ordering (Dickson's Lemma)
  - A^* with scattered subword ordering (Higman's Lemma) (for A wqo)

Prerun Embedding

Construct the ordering $\trianglelefteq$ over preruns inductively; recall

$$\mathsf{PreRuns}_{\mathbf{A}} \stackrel{\text{def}}{=} \mathbb{N}^d \times (\mathbb{N}^d \times \mathbf{A} \times \mathbb{N}^d)^* \times \mathbb{N}^d$$

Example (Run Embedding $\trianglelefteq$)



Lemma (Jančar, 1990)

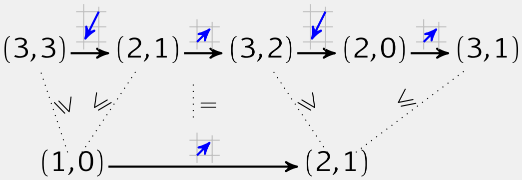
$(\mathsf{PreRuns}_{\mathbf{A}}, \trianglelefteq)$ *is a wqo.*

Prerun Embedding

Construct the ordering $\trianglelefteq$ over preruns inductively; recall

$$\mathsf{PreRuns}_{\mathbf{A}} \stackrel{\text{def}}{=} \mathbb{N}^d \times (\mathbb{N}^d \times \mathbf{A} \times \mathbb{N}^d)^* \times \mathbb{N}^d$$

Example (Run Embedding $\trianglelefteq$)



Lemma (Jančar, 1990)

$(\mathsf{PreRuns}_{\mathbf{A}}, \trianglelefteq)$ *is a wqo.*

Ideals as Canonical Bases

Lemma (Canonical Ideal Decomposition; Bonnet, 1975)

Every downward-closed subset $D \subseteq X$ of a wqo $(X, \leqslant)$ is the union of a unique finite family of incomparable (for the inclusion) ideals.

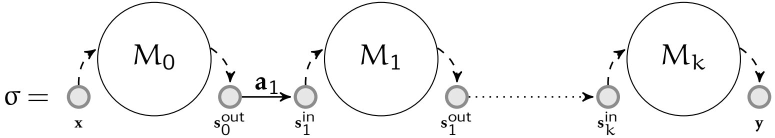
Finkel and Goubault-Larrecq (2009, 2012): effective representations of wqo ideals

Ideals as Canonical Bases

Lemma (Canonical Ideal Decomposition; Bonnet, 1975)
Every downward-closed subset $D \subseteq X$ of a wqo $(X, \leqslant)$ is the union of a unique finite family of incomparable (for the inclusion) ideals.

Finkel and Goubault-Larrecq (2009, 2012): effective representations of wqo ideals

Marked Graph Sequence (Ideal View)



A representation for (particular) prerun ideals with

$$I_\sigma \supseteq \downarrow\Omega_\sigma$$

Theorem (Perfectness as Ideal Adherence)
If σ is perfect then $I_\sigma = \downarrow\Omega_\sigma$.

KLMST Algorithm (Ideal View)

Construct a sequence $D_0, D_1, \ldots$ of downwards-closed sets, represented as finite sets of ideals, with $\forall n$

$$D_n \stackrel{\text{def}}{=} \bigcup_{\sigma \in S_n} I_\sigma \supseteq \downarrow \mathsf{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$$

init $D_0 \stackrel{\text{def}}{=} \mathsf{PreRuns}_{\mathbf{A}}$

$\forall n$
- if $D_n = I \sqcup D$ and $\neg\mathsf{perfect}(I)$,

  $D_{n+1} \stackrel{\text{def}}{=} D \sqcup \mathsf{decompose}(I)$

- otherwise stop:

  $D_n = \downarrow \mathsf{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$

KLMST Algorithm (Ideal View)

Construct a sequence $D_0, D_1, \ldots$ of downwards-closed sets, represented as finite sets of ideals, with $\forall n$

$$D_n \stackrel{\text{def}}{=} \bigcup_{\sigma \in S_n} I_\sigma \supseteq \downarrow \mathsf{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$$

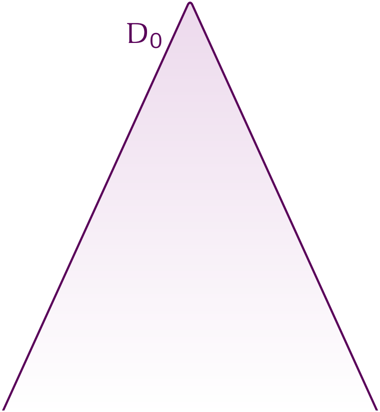
init $D_0 \stackrel{\text{def}}{=} \mathsf{PreRuns}_{\mathbf{A}}$

$\forall n$ ▸ if $D_n = I \sqcup D$ and perfect(I)

$D_{n+1} \stackrel{\text{def}}{=} D \sqcup \text{decompose}(I)$

▸ otherwise stop:

$D_n = \downarrow \mathsf{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$



KLMST Algorithm (Ideal View)

Construct a sequence $D_0, D_1, \ldots$ of downwards-closed sets, represented as finite sets of ideals, with $\forall n$

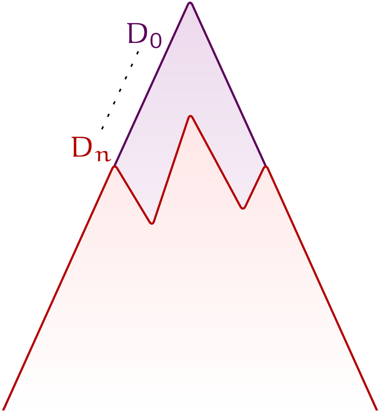
$$D_n \stackrel{\text{def}}{=} \bigcup_{\sigma \in S_n} I_\sigma \supseteq \downarrow \mathrm{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$$

init $D_0 \stackrel{\text{def}}{=} \mathrm{PreRuns}_{\mathbf{A}}$

$\forall n$
- if $D_n = I \sqcup D$ and $\neg\mathrm{perfect}(I)$,

  $D_{n+1} \stackrel{\text{def}}{=} D \cup \mathrm{decompose}(I)$
- otherwise stop:

  $D_n = \downarrow \mathrm{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$



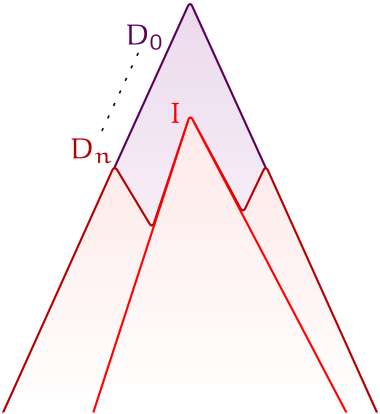
KLMST Algorithm (Ideal View)

Construct a sequence $D_0, D_1, \ldots$ of downwards-closed sets, represented as finite sets of ideals, with $\forall n$

$$D_n \stackrel{\text{def}}{=} \bigcup_{\sigma \in S_n} I_\sigma \supseteq \downarrow \mathsf{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$$

init $D_0 \stackrel{\text{def}}{=} \mathsf{PreRuns}_{\mathbf{A}}$

$\forall n$
- if $D_n = I \sqcup D$ and $\neg\mathrm{perfect}(I)$, $D_{n+1} \stackrel{\text{def}}{=} D \cup \mathrm{decompose}(I)$
- otherwise stop: $D_n = \downarrow \mathsf{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$



KLMST Algorithm (Ideal View)

Construct a sequence $D_0, D_1, \ldots$ of downwards-closed sets, represented as finite sets of ideals, with $\forall n$

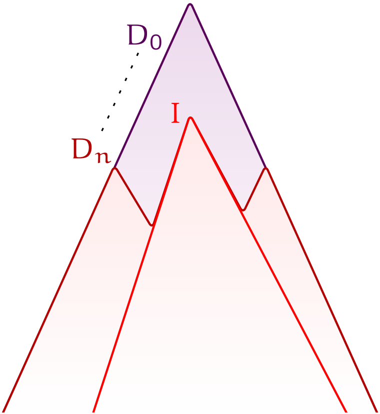
$$D_n \stackrel{\text{def}}{=} \bigcup_{\sigma \in S_n} I_\sigma \supseteq {\downarrow}\mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y})$$

init $D_0 \stackrel{\text{def}}{=} \mathrm{PreRuns}_{\mathbf{A}}$

$\forall n$
- if $D_n = I \sqcup D$ and $\neg\mathrm{perfect}(I)$,

  $D_{n+1} \stackrel{\text{def}}{=} D \cup \mathrm{decompose}(I)$
- otherwise stop:

  $D_n = {\downarrow}\mathrm{Runs}_{\mathbf{A}}(\mathbf{x},\mathbf{y})$



KLMST Algorithm (Ideal View)

Construct a sequence $D_0, D_1, \ldots$ of downwards-closed sets, represented as finite sets of ideals, with $\forall n$

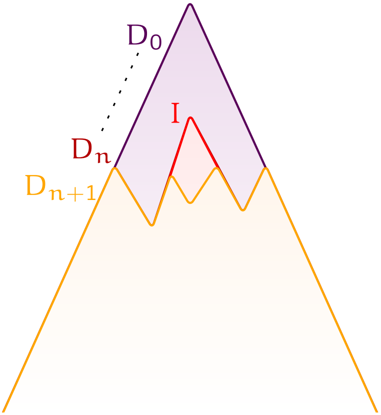
$$D_n \stackrel{\text{def}}{=} \bigcup_{\sigma \in S_n} I_\sigma \supseteq \downarrow \mathrm{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$$

init $D_0 \stackrel{\text{def}}{=} \mathrm{PreRuns}_{\mathbf{A}}$

$\forall n$
- if $D_n = I \sqcup D$ and $\neg\mathrm{perfect}(I)$,

  $D_{n+1} \stackrel{\text{def}}{=} D \cup \mathrm{decompose}(I)$
- otherwise stop:

  $D_n = \downarrow \mathrm{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$



KLMST Algorithm (Ideal View)

Construct a sequence $D_0, D_1, \ldots$ of downwards-closed sets, represented as finite sets of ideals, with $\forall n$

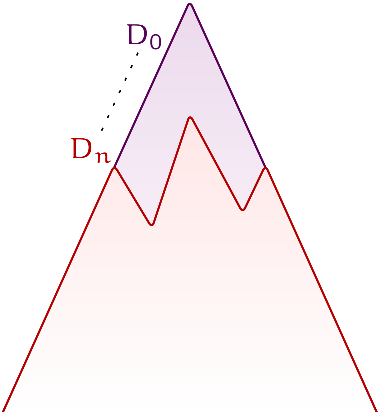
$$D_n \stackrel{\text{def}}{=} \bigcup_{\sigma \in S_n} I_\sigma \supseteq {\downarrow}\mathrm{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$$

init $D_0 \stackrel{\text{def}}{=} \mathrm{PreRuns}_{\mathbf{A}}$

$\forall n$
- if $D_n = I \sqcup D$ and $\neg\mathrm{perfect}(I)$,

  $D_{n+1} \stackrel{\text{def}}{=} D \cup \mathrm{decompose}(I)$
- otherwise stop:

  $D_n = {\downarrow}\mathrm{Runs}_{\mathbf{A}}(\mathbf{x}, \mathbf{y})$

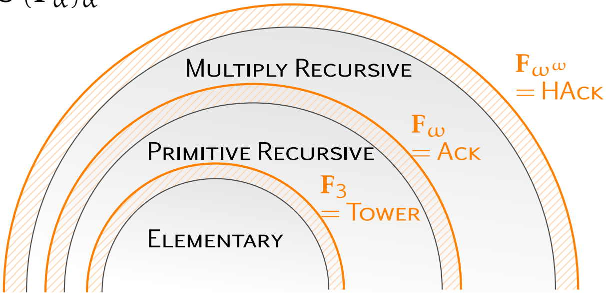


Mystery 2: Computational Complexity

Theorem (Upper Bound Theorem)

VAS Reachability is in $\mathbf{F}_{\omega^3}$.

- uses a length function theorem for ranking functions (S., 2014)
- ...which provides bounds in fast-growing complexity classes $(\mathbf{F}_\alpha)_\alpha$

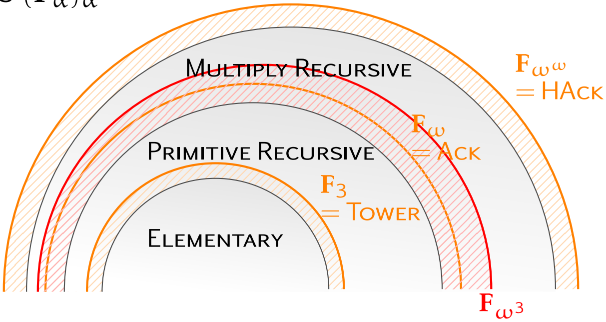


Mystery 2: Computational Complexity

Theorem (Upper Bound Theorem)

VAS Reachability is in $\mathbf{F}_{\omega^3}$.

- uses a length function theorem for ranking functions (S., 2014)
- ... which provides bounds in fast-growing complexity classes $(\mathbf{F}_\alpha)_\alpha$



Concluding Remarks

- ideals as an algorithmic tool to work with downward-closed sets
- new understanding of the KLMST decomposition
  extension to other models (BVASS, PDVAS,...)?
- immense complexity gap: ExpSpace vs. $\mathbf{F}_{\omega^3}$
  - only known tight case: PSpace-complete $d = 2$ + control states (as presented by Blondin et al., 2015, this morning)

Concluding Remarks

- ideals as an algorithmic tool to work with downward-closed sets
- new understanding of the KLMST decomposition extension to other models (BVASS, PDVAS,. . .)?
- immense complexity gap: ExpSpace vs. $\mathbf{F}_{\omega^3}$
  - only known tight case: PSpace-complete $d = 2$ + control states (as presented by Blondin et al., 2015, this morning)

Concluding Remarks

- ideals as an algorithmic tool to work with downward-closed sets
- new understanding of the KLMST decomposition extension to other models (BVASS, PDVAS,...)?
- immense complexity gap: ExpSpace vs. $\mathbf{F}_{\omega^3}$
  - only known tight case: PSpace-complete $d = 2$ + control states (as presented by Blondin et al., 2015, this morning)

Blondin, M., Finkel, A., Göller, S., Haase, C., and McKenzie, P., 2015. Reachability in two-dimensional vector addition systems with states is PSPACE-complete. In *Proc. LICS 2015*. IEEE Press. http://arxiv.org/abs/1412.4259. To appear.

Bonnet, R., 1975. On the cardinality of the set of initial intervals of a partially ordered set. In *Infinite and finite sets: to Paul Erdős on his 60th birthday, Vol. 1*, Coll. Math. Soc. János Bolyai, pages 189–198. North-Holland.

Figueira, D., Figueira, S., Schmitz, S., and Schnoebelen, Ph., 2011. Ackermannian and primitive-recursive bounds with Dickson's Lemma. In *LICS 2011*, pages 269–278. IEEE Press. doi:10.1109/LICS.2011.39.

Finkel, A. and Goubault-Larrecq, J., 2009. Forward analysis for WSTS, part I: Completions. In *Proc. STACS 2009*, volume 3 of *LIPIcs*, pages 433–444. LZI. doi:10.4230/LIPIcs.STACS.2009.1844.

Finkel, A. and Goubault-Larrecq, J., 2012. Forward analysis for WSTS, part II: Complete WSTS. *Logic. Meth. in Comput. Sci.*, 8(3:28):1–35. doi:10.2168/LMCS-8(3:28)2012.

Habermehl, P., Meyer, R., and Wimmel, H., 2010. The downward-closure of Petri net languages. In *Proc. ICALP 2010*, volume 6199 of *LNCS*, pages 466–477. Springer. doi:10.1007/978-3-642-14162-1_39.

Jančar, P., 1990. Decidability of a temporal logic problem for Petri nets. *Theor. Comput. Sci.*, 74(1):71–93. doi:10.1016/0304-3975(90)90006-4.

Karp, R.M. and Miller, R.E., 1969. Parallel program schemata. *Journal of Computer and System Sciences*, 3(2): 147–195. doi:10.1016/S0022-0000(69)80011-5.

Kosaraju, S.R., 1982. Decidability of reachability in vector addition systems. In *Proc. STOC'82*, pages 267–281. ACM. doi:10.1145/800070.802201.

Lambert, J.L., 1992. A structure to decide reachability in Petri nets. *Theor. Comput. Sci.*, 99(1):79–104. doi:10.1016/0304-3975(92)90173-D.

Leroux, J., 2011. Vector addition system reachability problem: a short self-contained proof. In *Proc. POPL 2011*, pages 307–316. ACM. doi:10.1145/1926385.1926421.

Lipton, R., 1976. The reachability problem requires exponential space. Technical Report 62, Yale University. http://cpsc.yale.edu/sites/default/files/files/tr63.pdf.

Mayr, E.W., 1981. An algorithm for the general Petri net reachability problem. In *Proc. STOC'81*, pages 238–246. ACM. doi:10.1145/800076.802477.

Müller, H., 1985. The reachability problem for VAS. In *Advances in Petri Nets 1984*, volume 188 of *LNCS*, pages 376–391. Springer. doi:10.1007/3-540-15204-0_21.

Sacerdote, G.S. and Tenney, R.L., 1977. The decidability of the reachability problem for vector addition systems. In *Proc. STOC'77*, pages 61–76. ACM. doi:10.1145/800105.803396.

Schmitz, S., 2014. Complexity bounds for ordinal-based termination. In *Proc. RP 2014*, volume 8762 of *LNCS*, pages 1–19. Springer. doi:10.1007/978-3-319-11439-2_1.

A Bit of Magic...

Theorem (Length Function Theorem, S., 2014)

(g,n)-controlled decreasing sequences $\alpha_0, \alpha_1, \ldots$ of ordinals $< \alpha$ with $n \geqslant |\alpha|$ are of length bounded by $g^\alpha(n)$ in the Hardy hierarchy.

Claim (KLMST control, using Figueira et al.*, 2011)*

In the KLMST algorithm, the sequence of ranks along any branch is controlled by $(H^{\omega^{d+1}}, |\mathbf{A}|)$.

As a result, the KLMST algorithm runs in space $\left(H^{\omega^{d+1}}\right)^{\omega^{\omega^3}}(|\mathbf{A}|)$, which is in $\mathbf{F}_{\omega^3}$.

A Bit of Magic: Controlled Sequences

Definition (Ordinal Norm)

For an ordinal $\alpha < \varepsilon_0$, define the *norm* $|\alpha|$ as the maximal coefficient appearing in its Cantor normal form $\alpha = \omega^{\alpha_1} \cdot c_1 + \cdots + \omega^{\alpha_n} \cdot c_n$:

$$|\alpha| \stackrel{\text{def}}{=} \max_{1 \leqslant j \leqslant n} (c_j, |\alpha_j|) .$$

Definition (Controlled Sequence)

Let $g\colon \mathbb{N} \to \mathbb{N}$ strictly monotone and $n \in \mathbb{N}$. A sequence of ordinals $\alpha_0, \alpha_1, \ldots$ is (g,n)-*controlled* if

$$\forall i \,.\, |\alpha_i| \leqslant g^i(n) .$$

In particular, $|\alpha_0| \leqslant n$.

A Bit of Magic: Hardy Functions

Ordinal-indexed hierarchy of functions $h^\alpha: \mathbb{N} \to \mathbb{N}$

Definition

Fix $h: \mathbb{N} \to \mathbb{N}$ strictly monotone:

$$h^0(x) \stackrel{\text{def}}{=} x \qquad h^{\alpha+1}(x) \stackrel{\text{def}}{=} h^\alpha(h(x)) \qquad h^\lambda(x) \stackrel{\text{def}}{=} h^{\lambda(x)}(x)$$

where $\lambda(0) < \lambda(1) < \cdots < \lambda$ is the standard fundamental sequence for the limit ordinal λ, e.g. $\omega(x) = x + 1$, $\omega^2(x) = \omega \cdot (x+1)$, $\omega^\omega(x) = \omega^{x+1}$.

Example

For instance for $H(x) \stackrel{\text{def}}{=} x + 1$:

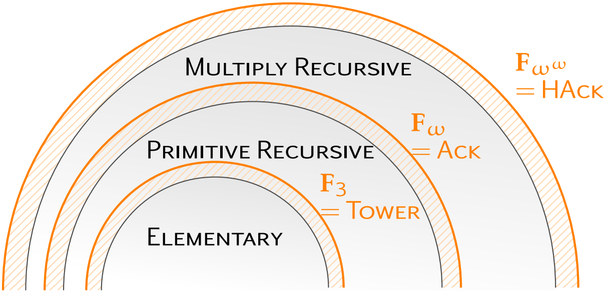
$$H^\omega(x) = 2x + 1 \qquad H^{\omega^2}(x) = 2^{x+1}(x+1) - 1$$

$$H^{\omega^3}(x) \approx 2^{\cdot^{\cdot^{\cdot^2}}}\Big\}x \text{ times} \qquad H^{\omega^\omega}(x) \approx \text{ackermann}(x)$$

A Bit of Magic: Fast-Growing Complexity

For $\alpha \geqslant 3$:

$$\mathscr{F}_{<\alpha} \stackrel{\text{def}}{=} \bigcup_{\beta<\omega^\alpha} \text{FDTime}(H^\beta(n)) \,, \quad \mathbf{F}_\alpha \stackrel{\text{def}}{=} \bigcup_{p\in\mathscr{F}_{<\alpha}} \text{DTime}(H^{\omega^\alpha}(p(n)))$$



A Bit of Magic: Fast-Growing Complexity

For $\alpha \geqslant 3$:

$$\mathscr{F}_{<\alpha} \stackrel{\text{def}}{=} \bigcup_{\beta<\omega^\alpha} \text{FDTime}(H^\beta(n)) , \quad \mathbf{F}_\alpha \stackrel{\text{def}}{=} \bigcup_{p\in\mathscr{F}_{<\alpha}} \text{DTime}(H^{\omega^\alpha}(p(n)))$$