Deciding Equivalence for Deterministic Top-down Tree-to-Weight Transducers

Malte Blattmann & Andreas Maletti

Universität Leipzig, Germany

RAQM — June 23, 2021

Personal Motivation

Connection

- Started PhD under supervision of Heiko Vogler in 2003 on

  weighted tree transducers

Personal Motivation

Connection

- Started PhD under supervision of Heiko Vogler in 2003 on

  weighted tree transducers

- Weighted tree transducers introduced by Werner in 1997

  Werner Kuich: *Formal Power Series over Trees.* Proc. DLT 1997

Personal Motivation

Connection

- Started PhD under supervision of Heiko Vogler in 2003 on

  weighted tree transducers

- Weighted tree transducers introduced by Werner in 1997

  Werner Kuich: *Formal Power Series over Trees.* Proc. DLT 1997

- Werner served as external reviewer of my thesis

The Power of Tree Series Transducers

Dissertation

zur Erlangung des akademischen Grades
Doctor rerum naturalium (Dr. rer. nat.)

vorgelegt an der
Technischen Universität Dresden
Fakultät Informatik

eingereicht von

Diplom-Informatiker Andreas Maletti
geboren am 22. November 1978 in Dresden

Gutachter:

Prof. Dr.-Ing. habil. *Heiko Vogler*
(Technische Universität Dresden)

Prof. Dr. rer. nat. habil. *Manfred Droste*
(Universität Leipzig)

O. Univ. Prof. Dr. phil. Dr. h.c. *Werner Kuich*
(Technische Universität Wien)

Tag der Verteidigung: 15. Juni 2006

Dresden, im Juli 2006

General Motivation

Weighted automata

- Offer elegant algebraic constructions (even for unweighted case)
- Enable use of powerful linear algebra
  (Hilbert's basis theorem, Gröbner bases)
- Yield more general results & better insights
- Standard model in several application areas (e.g. NLP)

Specific Motivation

Equivalence decidable

1. Deterministic Top-Down Tree-to-String Transducers
   [Seidl, Maneth, Kemper: J. ACM 2018]
2. Linear Det. Top-Down Tree Transducers with Output in Free Group
   [Löbel, Luttenberger, Seidl: Proc. DLT 2020]

Specific Motivation

Equivalence decidable

1. Deterministic Top-Down Tree-to-String Transducers
   [Seidl, Maneth, Kemper: J. ACM 2018]
2. Linear Det. Top-Down Tree Transducers with Output in Free Group
   [Löbel, Luttenberger, Seidl: Proc. DLT 2020]

Joint generalization

- Deterministic Top-Down Tree-to-Weight Transducer
- Weights in monoid $(S, \cdot, 1, 0)$ with adjoined zero 0
  ($s \cdot 0 = 0 = 0 \cdot s$ and $s \cdot s' = 0$ implies $0 \in \{s, s'\}$)

Specific Motivation

Equivalence decidable

1. Deterministic Top-Down Tree-to-String Transducers
   [Seidl, Maneth, Kemper: J. ACM 2018]
2. Linear Det. Top-Down Tree Transducers with Output in Free Group
   [Löbel, Luttenberger, Seidl: Proc. DLT 2020]

Joint generalization

- Deterministic Top-Down Tree-to-Weight Transducer
- Weights in monoid $(S, \cdot, 1, 0)$ with adjoined zero 0
  ($s \cdot 0 = 0 = 0 \cdot s$ and $s \cdot s' = 0$ implies $0 \in \{s, s'\}$)
- Free monoid $(A^*, \cdot, \varepsilon, \bot)$ in 1
- Free group $(G, \cdot, \varepsilon, \bot)$ in 2

Transducer Model

Fix $(S, \cdot, 1, 0)$ monoid with adjoined zero 0 and $[k] = \{1, \ldots, k\}$

$$X_{k,n} = \{x_{i,q} \mid i \in [k],\ q \in [n]\}$$

Definition (wDT transducer)

Deterministic top-down tree-to-weight transducer (wDT transducer) is tuple

$$(\Sigma, n, (\delta_k)_{k \in \mathbb{N}})$$

- ranked alphabet Σ of input symbols and number $n \in \mathbb{N}_+$ of states

Transducer Model

Fix $(S, \cdot, 1, 0)$ monoid with adjoined zero 0 and $[k] = \{1, \dots, k\}$

$$X_{k,n} = \{x_{i,q} \mid i \in [k],\ q \in [n]\}$$

Definition (wDT transducer)

Deterministic top-down tree-to-weight transducer (wDT transducer) is tuple

$$(\Sigma, n, (\delta_k)_{k \in \mathbb{N}})$$

- ranked alphabet Σ of input symbols and number $n \in \mathbb{N}_+$ of states
- transition functions $\delta_k \colon \Sigma_k \times [n] \to (X_{k,n} \cup S)^*$

Transducer Model

Let $S = \big(\{\alpha, \beta\}^* \times \mathbb{Z}_6\big) \cup \{\perp\}$ product monoid with adjoined $\perp$
(free monoid generated by $\{\alpha, \beta\}$ × additive group $\mathbb{Z}_6 = \mathbb{Z}/6\mathbb{Z}$)

Transducer Model

Let $S = \big(\{\alpha, \beta\}^* \times \mathbb{Z}_6\big) \cup \{\bot\}$ product monoid with adjoined $\bot$
(free monoid generated by $\{\alpha, \beta\}$ × additive group $\mathbb{Z}_6 = \mathbb{Z}/6\mathbb{Z}$)

Example transitions

$$\delta_2(\sigma, 1) = (\varepsilon, 1)\, x_{1,1}\, x_{1,3}\, x_{2,2} \qquad \delta_0(\alpha, 1) = \bot \qquad \delta_0(\beta, 1) = \bot$$
$$\delta_2(\sigma, 2) = (\varepsilon, 0)\, x_{1,2}\, x_{1,3}\, x_{2,2} \qquad \delta_0(\alpha, 2) = (\alpha, 0) \qquad \delta_0(\beta, 2) = (\beta, 0)$$
$$\delta_2(\sigma, 3) = (\varepsilon, 1)\, x_{1,3}\, x_{1,3} \qquad \delta_0(\alpha, 3) = (\varepsilon, 0) \qquad \delta_0(\beta, 3) = (\varepsilon, 0)$$

Transducer Model

Fix wDT transducer $\mathcal{M} = \big(\Sigma, n, (\delta_k)_{k \in \mathbb{N}}\big)$

Definition (state semantics)

For every state $q \in [n]$ define $[\![q]\!] \colon T_\Sigma \to S$ by

$$[\![q]\!]\big(\sigma(t_1, \ldots, t_k)\big) = [\![\delta_k(\sigma, q)]\!]^{t_1 \cdots t_k}$$

where $[\![\cdot]\!]^{t_1 \cdots t_k} \colon (X_{k,n} \cup S)^* \to S$ unique homomorphism given by

Transducer Model

Fix wDT transducer $\mathcal{M} = \big(\Sigma, n, (\delta_k)_{k \in \mathbb{N}}\big)$

Definition (state semantics)

For every state $q \in [n]$ define $[\![q]\!] \colon T_\Sigma \to S$ by

$$[\![q]\!]\big(\sigma(t_1, \ldots, t_k)\big) = [\![\delta_k(\sigma, q)]\!]^{t_1 \cdots t_k}$$

where $[\![\cdot]\!]^{t_1 \cdots t_k} \colon (X_{k,n} \cup S)^* \to S$ unique homomorphism given by

- $[\![x_{i,p}]\!]^{t_1 \cdots t_k} = [\![p]\!](t_i)$ for all $i \in [k]$ and $p \in [n]$

Transducer Model

Fix wDT transducer $\mathcal{M} = \big(\Sigma, n, (\delta_k)_{k\in\mathbb{N}}\big)$

Definition (state semantics)

For every state $q \in [n]$ define $[\![q]\!] \colon T_\Sigma \to S$ by

$$[\![q]\!]\big(\sigma(t_1, \dots, t_k)\big) = [\![\delta_k(\sigma, q)]\!]^{t_1 \cdots t_k}$$

where $[\![\cdot]\!]^{t_1 \cdots t_k} \colon (X_{k,n} \cup S)^* \to S$ unique homomorphism given by

- $[\![x_{i,p}]\!]^{t_1 \cdots t_k} = [\![p]\!](t_i)$ for all $i \in [k]$ and $p \in [n]$
- $[\![s]\!]^{t_1 \cdots t_k} = s$ for all $s \in S$

Transducer Model

Used transition

$$\delta_2(\sigma, 1) = (\varepsilon, 1)\, x_{1,2}\, x_{1,3}\, x_{2,2}$$

Applied transition

$$[\![1]\!]\Big(\sigma\big(\sigma(\alpha, \beta), \sigma(\beta, \alpha)\big) \Big) = (\varepsilon, 1) \cdot [\![2]\!]\big(\sigma(\alpha, \beta) \big) \cdot [\![3]\!]\big(\sigma(\alpha, \beta) \big) \cdot [\![2]\!]\big(\sigma(\beta, \alpha) \big)$$

Transducer Model

Definition (semantics)

Translation $\mathcal{M}\colon T_\Sigma \to S$ is $\mathcal{M}(t) = [\![1]\!](t)$ for all trees $t \in T_\Sigma$

Transducer Model

Transitions

$$
\begin{array}{lll}
\delta_2(\sigma,1) = (\varepsilon,1)\, x_{1,1}\, x_{1,3}\, x_{2,2} & \delta_0(\alpha,1) = \perp & \delta_0(\beta,1) = \perp \\
\delta_2(\sigma,2) = (\varepsilon,0)\, x_{1,2}\, x_{1,3}\, x_{2,2} & \delta_0(\alpha,2) = (\alpha,0) & \delta_0(\beta,2) = (\beta,0) \\
\delta_2(\sigma,3) = (\varepsilon,1)\, x_{1,3}\, x_{1,3} & \delta_0(\alpha,3) = (\varepsilon,0) & \delta_0(\beta,3) = (\varepsilon,0)
\end{array}
$$

Transducer Model

Transitions

$$
\begin{array}{lll}
\delta_2(\sigma,1) = (\varepsilon,1)\, x_{1,1}\, x_{1,3}\, x_{2,2} & \delta_0(\alpha,1) = \bot & \delta_0(\beta,1) = \bot \\
\delta_2(\sigma,2) = (\varepsilon,0)\, x_{1,2}\, x_{1,3}\, x_{2,2} & \delta_0(\alpha,2) = (\alpha,0) & \delta_0(\beta,2) = (\beta,0) \\
\delta_2(\sigma,3) = (\varepsilon,1)\, x_{1,3}\, x_{1,3} & \delta_0(\alpha,3) = (\varepsilon,0) & \delta_0(\beta,3) = (\varepsilon,0)
\end{array}
$$

Subderivations

$$
\begin{aligned}
[\![2]\!]\left(\begin{array}{c} \sigma \\ \alpha \quad \beta \end{array} \right) &= (\varepsilon,0) \cdot [\![2]\!](\alpha) \cdot [\![3]\!](\alpha) \cdot [\![2]\!](\beta) \\
&= (\varepsilon,0) \cdot (\alpha,0) \cdot (\varepsilon,0) \cdot (\beta,0) = (\alpha\beta,0)
\end{aligned}
$$

Transducer Model

Transitions

$$\delta_2(\sigma, 1) = (\varepsilon, 1)\, x_{1,1}\, x_{1,3}\, x_{2,2} \qquad \delta_0(\alpha, 1) = \bot \qquad \delta_0(\beta, 1) = \bot$$

$$\delta_2(\sigma, 2) = (\varepsilon, 0)\, x_{1,2}\, x_{1,3}\, x_{2,2} \qquad \delta_0(\alpha, 2) = (\alpha, 0) \qquad \delta_0(\beta, 2) = (\beta, 0)$$

$$\delta_2(\sigma, 3) = (\varepsilon, 1)\, x_{1,3}\, x_{1,3} \qquad \delta_0(\alpha, 3) = (\varepsilon, 0) \qquad \delta_0(\beta, 3) = (\varepsilon, 0)$$

Subderivations

$$\begin{aligned} [\![2]\!]\left(\begin{array}{c} \sigma \\ \alpha \quad \beta \end{array} \right) &= (\varepsilon, 0) \cdot [\![2]\!](\alpha) \cdot [\![3]\!](\alpha) \cdot [\![2]\!](\beta) \\ &= (\varepsilon, 0) \cdot (\alpha, 0) \cdot (\varepsilon, 0) \cdot (\beta, 0) = (\alpha\beta, 0) \end{aligned}$$

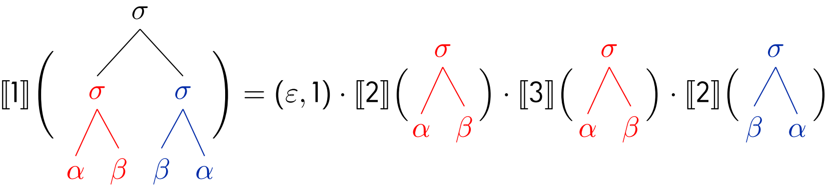
$$\begin{aligned} [\![3]\!]\left(\begin{array}{c} \sigma \\ \alpha \quad \beta \end{array} \right) &= (\varepsilon, 1) \cdot [\![3]\!](\alpha) \cdot [\![3]\!](\alpha) \\ &= (\varepsilon, 1) \cdot (\varepsilon, 0) \cdot (\varepsilon, 0) = (\varepsilon, 1) \end{aligned}$$

Transducer Model

$$[\![1]\!]\Big(\sigma\big(\sigma(\alpha, \beta), \sigma(\beta, \alpha)\big)\Big) = (\varepsilon, 1) \cdot [\![2]\!]\big(\sigma(\alpha, \beta)\big) \cdot [\![3]\!]\big(\sigma(\alpha, \beta)\big) \cdot [\![2]\!]\big(\sigma(\beta, \alpha)\big)$$

$$= (\varepsilon, 1) \cdot (\alpha\beta, 0) \cdot (\varepsilon, 1) \cdot (\beta\alpha, 0) = (\alpha\beta\beta\alpha, 2)$$

Transducer Model



$$= (\varepsilon, 1) \cdot (\alpha\beta, 0) \cdot (\varepsilon, 1) \cdot (\beta\alpha, 0) = (\alpha\beta\beta\alpha, 2)$$

Semantics

$$\mathcal{M}(t) = \big(\text{yield}(t),\ 2^{\text{left-spine}(t)} - \text{left-spine}(t) \bmod 6\big)$$

(left-spine(t) = number of σ-occurrences along left spine of t)

Setup

Definition

Two wDT transducers equivalent if their computed functions coincide

Definition

Two wDT transducers equivalent if their computed functions coincide

Assumption

- Finitely generated monoid $\langle E \rangle_S$ of used weights effectively embeds into multiplicative monoid of

  infinite, computable, and commutative field F

  (discussion when possible in second part)

Setup

Definition

Two wDT transducers equivalent if their computed functions coincide

Assumption

- Finitely generated monoid $\langle E \rangle_S$ of used weights effectively embeds into multiplicative monoid of

  infinite, computable, and commutative field F

  (discussion when possible in second part)

Approach following [Seidl, Maneth, Kemper 2018]

- Take disjoint union of both wDT transducers
- Prove invariance $[\![1]\!](t) - [\![n]\!](t) = 0$ for all $t \in T_\Sigma$ (1 and n initial states)

Invariance

Fix disjoint union wDT transducer $(\Sigma, n, (\delta_k)_{k \in \mathbb{N}})$

Definition (invariance)

Map $h \colon F^n \to F$ invariance if $h([\![t]\!]) = 0$ for all $t \in T_\Sigma$
where $[\![t]\!] = \big([\![1]\!](t), \dots, [\![n]\!](t)\big)$

<u>Note</u>

- Input wDT transducers equivalent iff $h^\star(s_1, \dots, s_n) = s_1 - s_n$ invariance

Invariance

Transitions (can be seen as monomials)

$$\delta_2(\sigma, 1) = (\varepsilon, 1)\, x_{1,2}\, x_{1,3}\, x_{2,2}$$
$$\delta_2(\sigma, 2) = (\varepsilon, 0)\, x_{1,2}\, x_{1,3}\, x_{2,2}$$
$$\delta_2(\sigma, 3) = (\varepsilon, 1)\, x_{1,3}\, x_{1,3}$$

Invariance

Transitions (can be seen as monomials)

$$\delta_2(\sigma, 1) = (\varepsilon, 1)\, x_{1,2}\, x_{1,3}\, x_{2,2}$$
$$\delta_2(\sigma, 2) = (\varepsilon, 0)\, x_{1,2}\, x_{1,3}\, x_{2,2}$$
$$\delta_2(\sigma, 3) = (\varepsilon, 1)\, x_{1,3}\, x_{1,3}$$

Polynomial reformulation

- Use tree vector notation $[\![\sigma(t_1, t_2)]\!]_1 = (\varepsilon, 1) \cdot [\![t_1]\!]_2 \cdot [\![t_1]\!]_3 \cdot [\![t_2]\!]_2$

Invariance

Transitions (can be seen as monomials)

$$\delta_2(\sigma, 1) = (\varepsilon, 1)\, x_{1,2}\, x_{1,3}\, x_{2,2}$$
$$\delta_2(\sigma, 2) = (\varepsilon, 0)\, x_{1,2}\, x_{1,3}\, x_{2,2}$$
$$\delta_2(\sigma, 3) = (\varepsilon, 1)\, x_{1,3}\, x_{1,3}$$

Polynomial reformulation

- Use tree vector notation $[\![\sigma(t_1, t_2)]\!]_1 = (\varepsilon, 1) \cdot [\![t_1]\!]_2 \cdot [\![t_1]\!]_3 \cdot [\![t_2]\!]_2$
- Group transitions $[\![\sigma]\!] = \big(\delta_2(\sigma, 1),\ \delta_2(\sigma, 2),\ \delta_2(\sigma, 3)\big)$

Invariance

Transitions (can be seen as monomials)

$$\delta_2(\sigma, 1) = (\varepsilon, 1)\, x_{1,2}\, x_{1,3}\, x_{2,2}$$
$$\delta_2(\sigma, 2) = (\varepsilon, 0)\, x_{1,2}\, x_{1,3}\, x_{2,2}$$
$$\delta_2(\sigma, 3) = (\varepsilon, 1)\, x_{1,3}\, x_{1,3}$$

Polynomial reformulation

- Use tree vector notation $[\![\sigma(t_1, t_2)]\!]_1 = (\varepsilon, 1) \cdot [\![t_1]\!]_2 \cdot [\![t_1]\!]_3 \cdot [\![t_2]\!]_2$
- Group transitions $[\![\sigma]\!] = \big(\delta_2(\sigma, 1),\ \delta_2(\sigma, 2),\ \delta_2(\sigma, 3)\big)$
- Thus generally

$$[\![\sigma(t_1, \dots, t_k)]\!] = [\![\sigma]\!] \big[x_{i,q} \leftarrow [\![t_i]\!]_q\big]$$

Invariance

Transitions

$$\delta_2(\sigma,1) = (\varepsilon,1)\, x_{1,1}\, x_{1,3}\, x_{2,2} \qquad \delta_0(\alpha,1) = \perp \qquad \delta_0(\beta,1) = \perp$$
$$\delta_2(\sigma,2) = (\varepsilon,0)\, x_{1,2}\, x_{1,3}\, x_{2,2} \qquad \delta_0(\alpha,2) = (\alpha,0) \qquad \delta_0(\beta,2) = (\beta,0)$$
$$\delta_2(\sigma,3) = (\varepsilon,1)\, x_{1,3}\, x_{1,3} \qquad \delta_0(\alpha,3) = (\varepsilon,0) \qquad \delta_0(\beta,3) = (\varepsilon,0)$$

Symbol semantics

$$[\![\sigma]\!] = \Big\langle (\varepsilon,1)\, x_{1,1}\, x_{1,3}\, x_{2,2},\ (\varepsilon,0)\, x_{1,2}\, x_{1,3}\, x_{2,2},\ (\varepsilon,1)\, x_{1,3}\, x_{1,3} \Big\rangle$$
$$[\![\alpha]\!] = \Big\langle \perp,\ (\alpha,0),\ (\varepsilon,0) \Big\rangle$$
$$[\![\beta]\!] = \Big\langle \perp,\ (\beta,0),\ (\varepsilon,0) \Big\rangle$$

Invariance

Definition

Ideal $I \subseteq F[X_n]$ inductive invariant if for all $\sigma \in \Sigma$

$$I \subseteq \left\{ p \in F[X_n] \;\middle|\; p\left[x_q \leftarrow \llbracket \sigma \rrbracket_q\right] \in \sum_{i=1}^{k} \left\langle I[x_q \leftarrow x_{i,q}] \right\rangle_{F[X_{k,n}]} \right\}$$

Invariance

Definition

Ideal $I \subseteq F[X_n]$ inductive invariant if for all $\sigma \in \Sigma$

$$I \subseteq \left\{ p \in F[X_n] \;\middle|\; p\left[x_q \leftarrow [\![\sigma]\!]_q\right] \in \sum_{i=1}^{k} \langle I[x_q \leftarrow x_{i,q}] \rangle_{F[X_{k,n}]} \right\}$$

Example

- Potential invariance $p = x_1 - x_3$
- Propagate symbol semantics

$$p[x_q \leftarrow [\![\sigma]\!]_q] = \Big((\varepsilon, 1)\, x_{1,1}\, x_{1,3}\, x_{2,2} \Big) - \Big((\varepsilon, 1)\, x_{1,3}\, x_{1,3} \Big)$$
$$p[x_q \leftarrow [\![\alpha]\!]_q] = \Big(\perp \Big) - \Big((\varepsilon, 0) \Big)$$

Invariance

Theorem [Seidl, Maneth, Kemper 2018]

Let I be inductive invariant and $\widetilde{I}$ set of all invariances

1. Every $p \in I$ is invariance

Theorem [Seidl, Maneth, Kemper 2018]

Let I be inductive invariant and $\widetilde{I}$ set of all invariances

1. Every $p \in I$ is invariance
2. $\widetilde{I}$ is inductive invariant

Invariance

Theorem

Each ideal $I \subseteq F[X_n]$ finitely generated and ideals recursively enumerable

Invariance

Theorem

Each ideal $I \subseteq F[X_n]$ finitely generated and ideals recursively enumerable

Proof.

Hilbert's basis theorem proves first statement and enumeration of finite sets $P \subseteq F[X_n]$ generating ideals using recursive enumeration of F $\square$

Deciding Equivalence

Semidecision algorithm for equivalence

1. Select next $P \subseteq F[X_n]$

Deciding Equivalence

Semidecision algorithm for equivalence

1. Select next $P \subseteq F[X_n]$
2. If $I = \langle P \rangle_{F[X_n]}$ inductive invariant <u>and</u> $h^\star \in I$, then return **yes**

Deciding Equivalence

Semidecision algorithm for equivalence

1. Select next $P \subseteq F[X_n]$
2. If $I = \langle P \rangle_{F[X_n]}$ inductive invariant <u>and</u> $h^\star \in I$, then return **yes**
3. Back to 1

Deciding Equivalence

Semidecision algorithm for equivalence

1. Select next $P \subseteq F[X_n]$
2. If $I = \langle P \rangle_{F[X_n]}$ inductive invariant <u>and</u> $h^\star \in I$, then return **yes**
3. Back to 1

Semidecision algorithm for <u>non-equivalence</u>

1. Select next $t \in T_\Sigma$

Deciding Equivalence

Semidecision algorithm for equivalence

1. Select next $P \subseteq F[X_n]$
2. If $I = \langle P \rangle_{F[X_n]}$ inductive invariant and $h^\star \in I$, then return **yes**
3. Back to 1

Semidecision algorithm for non-equivalence

1. Select next $t \in T_\Sigma$
2. If $[\![1]\!](t) \neq [\![n]\!](t)$, then return **yes**

Deciding Equivalence

Semidecision algorithm for equivalence

1. Select next $P \subseteq F[X_n]$
2. If $I = \langle P \rangle_{F[X_n]}$ inductive invariant and $h^\star \in I$, then return **yes**
3. Back to 1

Semidecision algorithm for non-equivalence

1. Select next $t \in T_\Sigma$
2. If $[\![1]\!](t) \neq [\![n]\!](t)$, then return **yes**
3. Back to 1

Deciding Equivalence

Monoid $\langle E \rangle_S$ of used weights effectively embeds into multiplicative monoid of infinite, computable, and commutative field F

Theorem

Equivalence of wDT transducers is decidable

Deciding Equivalence

Monoid $\langle E \rangle_S$ of used weights effectively embeds into multiplicative monoid of infinite, computable, and commutative field F

Theorem

Equivalence of wDT transducers is decidable

Proof.

Two trivially correct semidecidability algorithms yield decidability.
Testing inductive invariant and membership of $h^\star$ effective via Gröbner basis of I $\square$

Embedding Monoids into Fields

Requirements for monoid $(S, \cdot, 1)$

- Finitely presented (for effective computation)

Embedding Monoids into Fields

Requirements for monoid $(S, \cdot, 1)$

- Finitely presented (for effective computation)
- Commutative: $s \cdot s' = s' \cdot s$ (necessary)

Embedding Monoids into Fields

Requirements for monoid $(S, \cdot, 1)$

- Finitely presented (for effective computation)
- Commutative: $s \cdot s' = s' \cdot s$ (necessary)
- Cancellative: $s \cdot s_1 = s \cdot s_2$ implies $s_1 = s_2$ (necessary)

Embedding Monoids into Fields

Requirements for monoid $(S, \cdot, 1)$

- Finitely presented (for effective computation)
- Commutative: $s \cdot s' = s' \cdot s$ (necessary)
- Cancellative: $s \cdot s_1 = s \cdot s_2$ implies $s_1 = s_2$ (necessary)
- Locally cyclic torsion subgroup (necessary)

Embedding Monoids into Fields

Requirements for monoid $(S, \cdot, 1)$

- Finitely presented (for effective computation)
- Commutative: $s \cdot s' = s' \cdot s$ (necessary)
- Cancellative: $s \cdot s_1 = s \cdot s_2$ implies $s_1 = s_2$ (necessary)
- Locally cyclic torsion subgroup (necessary)
  - Torsion subgroup $S_0 = \{s \in S \mid \exists n \in \mathbb{N}_+ \colon s^n = 1\}$
    (subgroup due to cancellation)

Embedding Monoids into Fields

Requirements for monoid $(S, \cdot, 1)$

- Finitely presented (for effective computation)
- Commutative: $s \cdot s' = s' \cdot s$ (necessary)
- Cancellative: $s \cdot s_1 = s \cdot s_2$ implies $s_1 = s_2$ (necessary)
- Locally cyclic torsion subgroup (necessary)
  - Torsion subgroup $S_0 = \{s \in S \mid \exists n \in \mathbb{N}_+ \colon s^n = 1\}$
    (subgroup due to cancellation)
  - Locally cyclic group G: every finitely generated subgroup of G is cyclic

Embedding Monoids into Fields

Example monoid $(\{a\}^*, \cdot, \varepsilon)$ (isomorphic to $(\mathbb{N}, +, 0)$)

- Free monoid generated by a thus finitely presented
- Commutative since $a^m a^n = a^n a^m$
- Cancellative since $a^m a^n = a^m a^k$ implies $a^n = a^k$
- Trivial torsion subgroup $\{\varepsilon\}$ since $(a^m)^n \neq \varepsilon$ for all $n \in \mathbb{N}_+$ if $m \neq 0$

Embedding Monoids into Fields

Example monoid $(\{a\}^*, \cdot, \varepsilon)$ (isomorphic to $(\mathbb{N}, +, 0)$)

- Free monoid generated by a thus finitely presented
- Commutative since $a^m a^n = a^n a^m$
- Cancellative since $a^m a^n = a^m a^k$ implies $a^n = a^k$
- Trivial torsion subgroup $\{\varepsilon\}$ since $(a^m)^n \neq \varepsilon$ for all $n \in \mathbb{N}_+$ if $m \neq 0$

Example monoid $\{a\}^* \times \mathbb{Z}_6$

- Finitely presented $\langle a, 1 \mid a1 = 1a, 1^6 \rangle$
- Commutative since product of commutative monoids
- Cancellative since product of cancellative monoids (group $\mathbb{Z}_6$)
- Torsion subgroup $\{\varepsilon\} \times \mathbb{Z}_6 \simeq \mathbb{Z}_6$ cyclic

Embedding Monoids into Fields

Embed monoid S into group

- Utilize Grothendieck group G
  (embed additive monoid of non-negative integers into integers or multiplicative monoid of integers into rationals)
- G finitely presented
  $\langle E \cup E^{-1} \mid ee' = e \cdot e', \Pi \rangle$ for finite presentation $\langle E, \Pi \rangle$ of S
- Monoid S embeds into G due to cancellation
- Torsion subgroups of G and S isomorphic $(G_0 \simeq S_0)$

$F^\star$ multiplicative group of field F and additive group $\mathbb{Z}_n = \mathbb{Z}/n\mathbb{Z}$

Theorem

Every finitely presented Abelian group with locally cyclic torsion subgroup effectively embeds into $F^\star$ of infinite, computable, and commutative field F

Embedding Monoids into Fields

$F^\star$ multiplicative group of field F and additive group $\mathbb{Z}_n = \mathbb{Z}/n\mathbb{Z}$

Theorem

Every finitely presented Abelian group with locally cyclic torsion subgroup effectively embeds into $F^\star$ of infinite, computable, and commutative field F

Proof sketch (1/2).

Wlog. suppose that finitely presented Abelian group G has non-trivial 2-component in torsion subgroup G_0.

Embedding Monoids into Fields

$F^\star$ multiplicative group of field F and additive group $\mathbb{Z}_n = \mathbb{Z}/n\mathbb{Z}$

Theorem

Every finitely presented Abelian group with locally cyclic torsion subgroup <u>effectively</u> embeds into $F^\star$ of infinite, computable, and commutative field F

Proof sketch (1/2).

Wlog. suppose that finitely presented Abelian group G has non-trivial 2-component in torsion subgroup G_0. Invoke invariant factor decomposition of fundamental theorem for finitely generated Abelian groups to obtain

$$G \simeq \mathbb{Z}^r \times \mathbb{Z}_{k_1} \times \cdots \times \mathbb{Z}_{k_n}$$

with $1 < k_1 < \cdots < k_n$ and $k_i | k_{i+1}$ for all $1 \leq i < n$

Proof sketch (2/2).

Torsion subgroup is $G_0 \simeq \{0\}^r \times \mathbb{Z}_{k_1} \times \cdots \times \mathbb{Z}_{k_n}$ and thus finite and finitely generated.

Proof sketch (2/2).

Torsion subgroup is $G_0 \simeq \{0\}^r \times \mathbb{Z}_{k_1} \times \cdots \times \mathbb{Z}_{k_n}$ and thus finite and finitely generated. By assumption G_0 cyclic.

Proof sketch (2/2).

Torsion subgroup is $G_0 \simeq \{0\}^r \times \mathbb{Z}_{k_1} \times \cdots \times \mathbb{Z}_{k_n}$ and thus finite and finitely generated. By assumption G_0 cyclic. Since no $k, \ell \in \{k_1, \ldots, k_n\}$ co-prime, no product $\mathbb{Z}_k \times \mathbb{Z}_\ell$ cyclic.

Proof sketch (2/2).

Torsion subgroup is $G_0 \simeq \{0\}^r \times \mathbb{Z}_{k_1} \times \cdots \times \mathbb{Z}_{k_n}$ and thus finite and finitely generated. By assumption G_0 cyclic. Since no $k, \ell \in \{k_1, \ldots, k_n\}$ co-prime, no product $\mathbb{Z}_k \times \mathbb{Z}_\ell$ cyclic. Hence $n = 1$ and $G \simeq \mathbb{Z}^r \times \mathbb{Z}_k$ with k even.

Proof sketch (2/2).

Torsion subgroup is $G_0 \simeq \{0\}^r \times \mathbb{Z}_{k_1} \times \cdots \times \mathbb{Z}_{k_n}$ and thus finite and finitely generated. By assumption G_0 cyclic. Since no $k, \ell \in \{k_1, \ldots, k_n\}$ co-prime, no product $\mathbb{Z}_k \times \mathbb{Z}_\ell$ cyclic. Hence $n = 1$ and $G \simeq \mathbb{Z}^r \times \mathbb{Z}_k$ with k even. Embed into cyclotomic extension field $F = \mathbb{Q}(\zeta_k)$ with primitive k-th root ζ_k of unity.

Embedding Monoids into Fields

Proof sketch (2/2).

Torsion subgroup is $G_0 \simeq \{0\}^r \times \mathbb{Z}_{k_1} \times \cdots \times \mathbb{Z}_{k_n}$ and thus finite and finitely generated. By assumption G_0 cyclic. Since no $k, \ell \in \{k_1, \ldots, k_n\}$ co-prime, no product $\mathbb{Z}_k \times \mathbb{Z}_\ell$ cyclic. Hence $n = 1$ and $G \simeq \mathbb{Z}^r \times \mathbb{Z}_k$ with k even. Embed into cyclotomic extension field $F = \mathbb{Q}(\zeta_k)$ with primitive k-th root ζ_k of unity. Then torsion subgroup F_0 of F isomorphic to G_0 and

$$(z_1, \ldots, z_r, t) \mapsto \zeta_k^t \cdot \prod_{i=1}^{r} p_i^{z_i} \qquad \text{for all } z_1, \ldots, z_r \in \mathbb{Z},\ t \in \mathbb{Z}_k$$

effective embedding into $F^\star$ with first r primes $p_1, \ldots, p_r$ $\square$

Final notes

- Torsion subgroup of $F^\star$ locally cyclic for every commutative field F
  [Cohn: Bemerkung über multiplikative Gruppe eines Körpers. Archiv der Mathematik]
  [May: Multiplicative groups of fields. Proc. London Mathematical Society]
- $F^\star$ trivially commutative & cancellative

Deciding Equivalence

Theorem

Equivalence is decidable for wDT transducers over finitely presented, cancellative, and commutative monoids with locally cyclic torsion subgroup

Theorem

Equivalence is decidable for wDT transducers over finitely presented, cancellative, and commutative monoids with locally cyclic torsion subgroup

Thank you for the attention