Cost Register Automata

2019.12.20

Overview

Contents

Introduction

A Cost Register Automaton (CRA)

- process a data word (a sequence of tagged values) and outputs a value in a data set that is computed using a given set of operations over the data set;
- provides a machine-based characterization of the class of *regular transductions* from strings to costs.

Contents

Data Transductions

Notations

- Σ is a finite alphabet whose elements are called tags;
- D is a data set of data values.

Data Transductions

Notations

- Σ is a finite alphabet whose elements are called tags;
- D is a data set of data values.

Definition

A data word is a sequence of tagged values, i.e. a word over the alphabet $\Sigma \times D$.

Data Transductions

Notations

- Σ is a finite alphabet whose elements are called tags;
- D is a data set of data values.

Definition

A data word is a sequence of tagged values, i.e. a word over the alphabet $\Sigma \times D$.

Example

Suppose $\Sigma = \{a, \#\}, D = \mathbb{N}$. Then

$$(a, 0)(a, 2)(\#, 0)(a, 1)(\#, 2)$$

is a data word.

Data Transductions

Definition

A data transduction is a partial function of type

$$(\Sigma \times D)^* \rightharpoonup D.$$

Data Transductions

Definition

A data transduction is a partial function of type

$$(\Sigma \times D)^* \rightharpoonup D.$$

Definition

A multi-valued transduction is a partial function of type

$$(\Sigma \times D)^* \to \mathcal{P}(D),$$

where $\mathcal{P}$ is the powerset operator.

Data Transductions

Definition

For a data word $w \in (\Sigma \times D)^*$, $w|_\Sigma$ denotes the elementwise projection of w to the tag component. Formally,

$$(a_1, d_1)(a_2, d_2) \cdots (a_n, d_n)|_\Sigma = a_1 a_2 \cdots a_n$$

Data Transductions

Definition

For a data word $w \in (\Sigma \times D)^*$, $w|_\Sigma$ denotes the elementwise projection of w to the tag component. Formally,

$$(a_1, d_1)(a_2, d_2) \cdots (a_n, d_n)|_\Sigma = a_1 a_2 \cdots a_n$$

Definition

The rate of f is the language $\mathsf{R}(f) \subseteq \Sigma^*$ defined as follows:

$$\mathsf{R}(f) = \{\sigma \in \Sigma^* | \forall w \text{ with } w|_\Sigma = \sigma, f(w) \text{ is defined}\}.$$

Contents

Notations

- X is a set of variables;
- $\mathcal{O}$ is a family of constants and operations that are allowed on the data set D;
  - $\mathcal{O}_n$ denotes the set of n-ary operations that are contained in $\mathcal{O}$;
  - $\mathcal{O}_0$ is the set of constants in $\mathcal{O}$, in particular.

Syntax and Semantics of CRAs

Notations

- X is a set of variables;
- $\mathcal{O}$ is a family of constants and operations that are allowed on the data set D;
  - $\mathcal{O}_n$ denotes the set of n-ary operations that are contained in $\mathcal{O}$;
  - $\mathcal{O}_0$ is the set of constants in $\mathcal{O}$, in particular.

Definition

$\mathbb{E}_{\mathcal{O}}[X]$ denotes the set of expressions over X, defined by the following rules:

- $c \in \mathbb{E}_{\mathcal{O}}[X]$, if constanc $c \in \mathcal{O}$;
- $x \in \mathbb{E}_{\mathcal{O}}[X]$, if variable $x \in X$;
- $op(t_1, \ldots, t_n) \in \mathbb{E}_{\mathcal{O}}[x]$, if n-ary operation $op \in \mathcal{O}$, $t_i \in \mathbb{E}_{\mathcal{O}}[X]$ for all $i = 1, \ldots, n$.

Syntax and Semantics of CRAs

Definition (CRA)

A (nondeterministic, copyful) Cost Register Automaton (NCRA) over the tag alphabet Σ, data values D, and data operations $\mathcal{O}$ is a tuple

$$\mathcal{A} = (Q, X, \Delta, I, F),$$

where

- Q is a finite set of states,
- X is a finite set of registers,
- $\Delta \subseteq Q \times \Sigma \times U_{\mathcal{O}} \times Q$ is the set of transitions with $U_{\mathcal{O}}$ the set of register updates $X \to \mathbb{E}_{\mathcal{O}}[X \cup \{\mathtt{val}\}]$,
- $I : Q \rightharpoonup (X \to \mathbb{E}_{\mathcal{O}}[\emptyset])$ is the initialization function, and
- $F : Q \rightharpoonup \mathbb{E}_{\mathcal{O}}[X]$ is the finalization function.

Syntax and Semantics of CRAs

Branches

abbreviations	NCRA	UCRA	DCRA
the automaton is	nondeterministic	unambiguous	deterministic

Syntax and Semantics of CRAs

Branches

abbreviations	NCRA	UCRA	DCRA
the automaton is	nondeterministic	unambiguous	deterministic

Definition

A CRA is said to be copyless if

1. $\forall (p, a, \theta, q) \in \Delta, \forall x \in X$, there is at most one occurrence of x in the list of expressions $\theta(x_1), \ldots, \theta(x_n)$, where $x_1, \ldots, x_n$ is an enumeration of X, and
2. $\forall q \in \mathrm{dom}(F), \forall x \in X$, there is at most one occurrence of x in the expression $F(q)$.

Definition

A function $\alpha : X \to D$ is called a variable assignment. It extends uniquely to a homomorphism $\hat{\alpha} : \mathbb{E}_{\mathcal{O}}[X] \to D$.

Syntax and Semantics of CRAs

Definition

A function $\alpha : X \to D$ is called a variable assignment. It extends uniquely to a homomorphism $\hat{\alpha} : \mathbb{E}_{\mathcal{O}}[X] \to D$.

Notations

- `val` is a special symbol, referring to the value of the current data item;
- $\alpha[\texttt{val} \mapsto d]$ means the extension of α that maps `val` to d.

Syntax and Semantics of CRAs

Definition

A function $\alpha : X \to D$ is called a variable assignment. It extends uniquely to a homomorphism $\hat{\alpha} : \mathbb{E}_{\mathcal{O}}[X] \to D$.

Notations

- $\texttt{val}$ is a special symbol, referring to the value of the current data item;
- $\alpha[\texttt{val} \mapsto d]$ means the extension of α that maps $\texttt{val}$ to d.

Definition

An expression $t \in \mathbb{E}_{\mathcal{O}}[X]$ denotes a function $[\![t]\!] : D^X \to D$, defined as:

for a variable assignment $\alpha : X \to D$, put $[\![t]\!](\alpha) = \hat{\alpha}(t)$.

Syntax and Semantics of CRAs

Definition

A function $\alpha : X \to D$ is called a variable assignment. It extends uniquely to a homomorphism $\hat{\alpha} : \mathbb{E}_{\mathcal{O}}[X] \to D$.

Notations

- $\mathtt{val}$ is a special symbol, referring to the value of the current data item;
- $\alpha[\mathtt{val} \mapsto d]$ means the extension of α that maps $\mathtt{val}$ to d.

Definition

An expression $t \in \mathbb{E}_{\mathcal{O}}[X]$ denotes a function $[\![t]\!] : D^X \to D$, defined as:

for a variable assignment $\alpha : X \to D$, *put* $[\![t]\!](\alpha) = \hat{\alpha}(t)$.

An expression $t \in \mathbb{E}_{\mathcal{O}}[X \cup \{\mathtt{val}\}]$ denotes a function $[\![t]\!] : D^X \times D \to D$,

$$[\![t]\!](\alpha, d) = \hat{\beta}(t) \text{ where } \beta = \alpha[\mathtt{val} \mapsto d] : X \cup \{\mathtt{val}\} \to D$$

Syntax and Semantics of CRAs

Definition

For an input sequence $w = (a_1, d_1)(a_2, d_2) \cdots (a_n, d_n) \in (\Sigma \times D)^*$, we define a w-run in $\mathcal{A}$ to be a sequence

$$(q_0, \alpha_0) \xrightarrow{(a_1,d_1)} (q_1, \alpha_1) \xrightarrow{(a_2,d_2)} (q_2, \alpha_2) \xrightarrow{(a_3,d_3)} \cdots \xrightarrow{(a_n,d_n)} (q_n, \alpha_n)$$

with $q_i \in Q$ and $\alpha_i : X \to D$ for all i so that:

1. *Initialization:* $q_0 \in \mathrm{dom}(I)$ and $\alpha_0(x) = [\![I(q_0)(x)]\!]$ for every register $x \in X$;
2. *Transition:* $\forall (p, \alpha) \xrightarrow{(a,d)} (q, \beta)$, $\exists \theta \in U_{\mathcal{O}}$ with $(p, a, \theta, q) \in \Delta$ s.t.

$$\forall x \in X,\ \beta(x) = [\![\theta(x)]\!](\alpha, d);$$

3. *Finalization:* $q_n \in \mathrm{dom}(F)$.

The value of the run is $[\![F(q_n)]\!](\alpha_n)$.

Contents

Example 1

Suppose

$\Sigma = \{a, b\}, D = \mathbb{N}, \mathcal{O} = \{0, +\}$

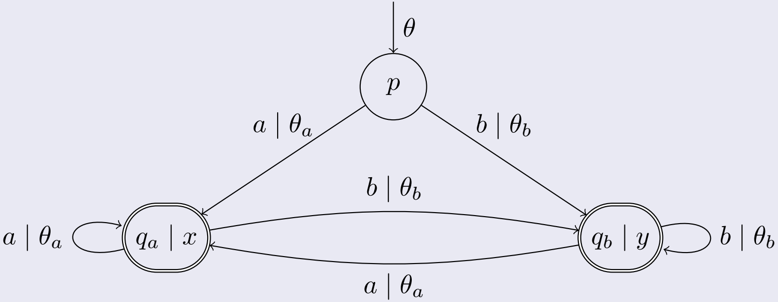
Transduction

$$f : (\Sigma \times D)^* \rightharpoonup D, \text{ defined on all nonempty sequences.}$$

If a sequence ends with an a-labeled (b-labeled) value, then f outputs the sum of all a-labeled (b-labeled) values in the sequence.

Example 1

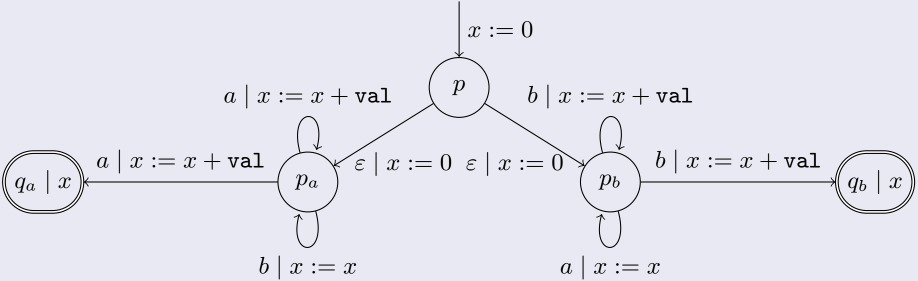
Copyless DCRA



$$\theta = \begin{cases} x := 0 \\ y := 0 \end{cases} \qquad \theta_a = \begin{cases} x := x + \texttt{val} \\ y := y \end{cases} \qquad \theta_b = \begin{cases} x := x \\ y := y + \texttt{val} \end{cases}$$

Example 2

Copyless UCRA



Example 3

Suppose

$\Sigma = \{a, \#\}$, $D = \mathbb{N}$, $\mathcal{O} = \{0, +, \max\}$.

Transduction

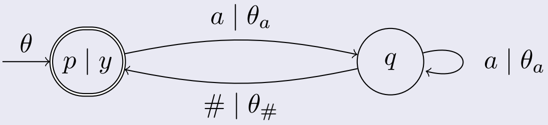
$$f : (\Sigma \times D)^* \rightharpoonup D, \quad \text{rate: } (a^+\#)^*$$

f outputs the maximum cost over all input blocks, where

- a block is a maximal subsequence of the input that is of the form $aa \ldots a\#$,
- the cost of a block is the sum of the a-labeled values.

Example 3

Copyless DCRA



$$\theta = \begin{cases} x := 0 \\ y := 0 \end{cases} \qquad \theta_a = \begin{cases} x := x + \texttt{val} \\ y := y \end{cases} \qquad \theta_\# = \begin{cases} x := 0 \\ y := \max(y, x) \end{cases}$$

Example 4

Suppose

$\Sigma = \{a, b\}, D = \mathbb{N}, \mathcal{O} = \{0, \ominus\}$. $x \ominus y = \max(x - y, 0)$.

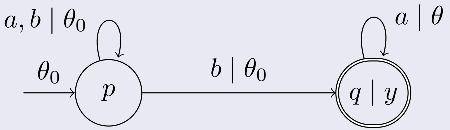
Transduction

$$f : (\Sigma \times D)^* \rightharpoonup D,$$

defined on sequences that contain at least one a-labeled value.
f outputs the maximum drawdown in the input signal after the last occurrence of a b-labeled value.

Example 4

Copyful UCRA



$$\theta_0 = \begin{cases} x := 0 \\ y := 0 \end{cases} \qquad \theta = \begin{cases} x := \max(x, \mathtt{val}) \\ y := \max(y, \max(x, \mathtt{val}) \ominus \mathtt{val}) \end{cases}$$

Example 5

Suppose

$\Sigma = \{a\}, D = \mathbb{Q}, \mathcal{O} = \{0, op\}$.

$$op(x, y) = \lambda \cdot x + y, \quad \lambda \in (0, 1).$$

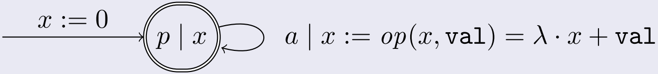
Transduction

For w with $w|_D = d_1 d_2 \dots d_n \in D^*$, f outputs

$$\lambda^{n-1} \cdot d_1 + \cdots + \lambda \cdot d_{n-1} + d_n.$$

Example 5

CRA



Example 5

Transduction

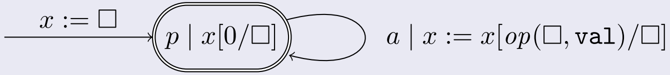
$$g(w_1 w_2 \dots w_n) = f(w_n \dots w_2 w_1).$$

Example 5

Transduction

$$g(w_1 w_2 \dots w_n) = f(w_n \dots w_2 w_1).$$

CRA with a hole



$x[t/\square]$ denotes the result of substituting t for $\square$ in the term that x holds.