NETWORKS OF EVOLUTIONARY PICTURE PROCESSORS

Victor Mitrana

Faculty of Mathematics and Computer Science

University of Bucharest, Romania

and

Department of Information Systems and Computation

Technical University of Valencia

mitrana@fmi.unibuc.ro

GENERAL IDEA

A set of nodes that are connected: a **network**
Each node: **evolutionary**/**splicing processor**
Two distinguished nodes: **In** and **Out**

The edges: **bidirectional communication channels**
Information: **rectangular pictures**
Restricted communication by: **input/output filters.**

Input: a string in ***In***.
Computation: Evolution/Splicing, Communication, ...
Halting: a string enters ***Out*** or two consecutive identical configurations.
Acceptance: a picture enters ***Out***

PICTURES

π =

a	b	b	a	c
b	b	a	a	b
c	b	a	a	a
b	b	a	a	a

ρ =

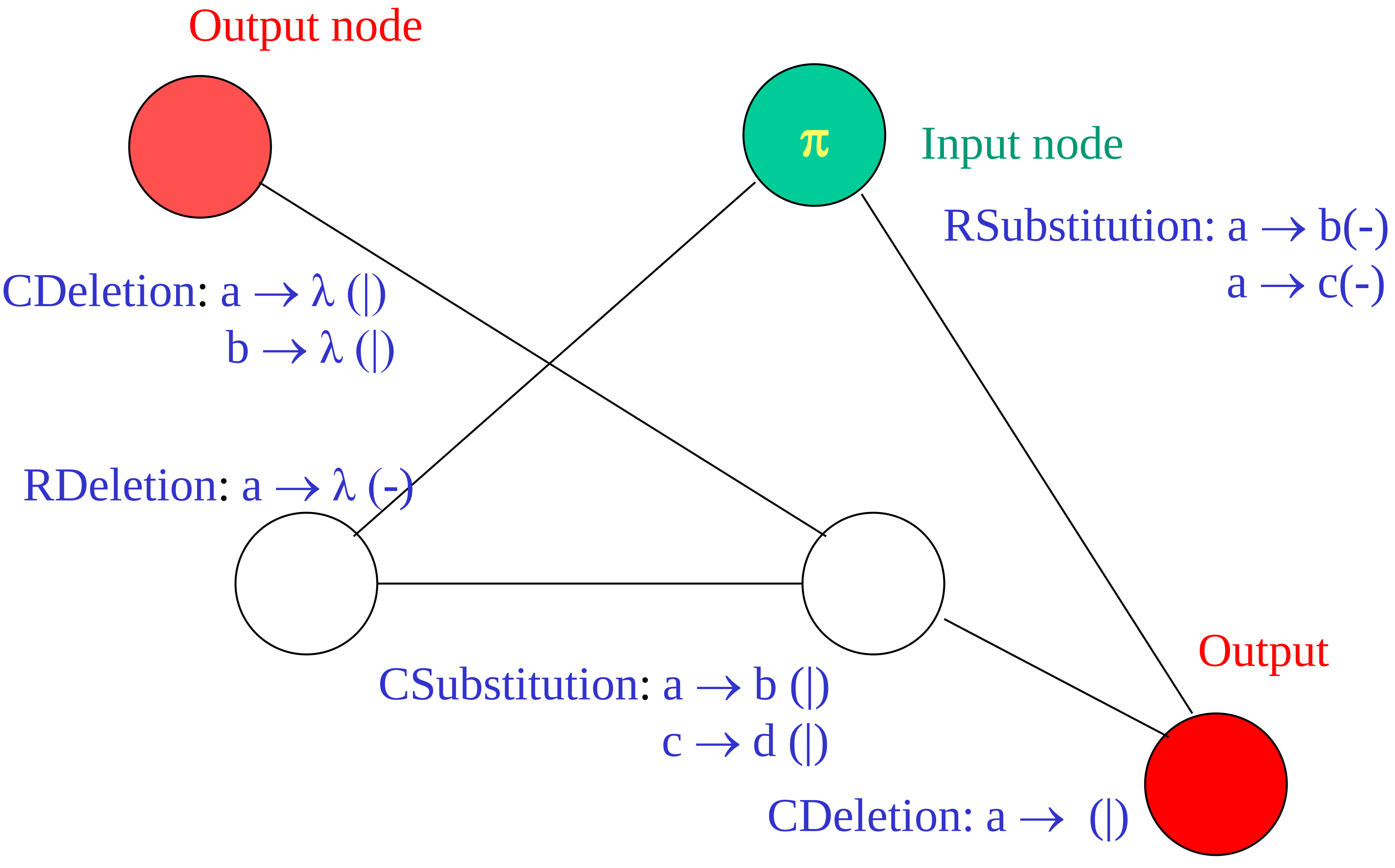
a	b	b	b	b	a	b
b	b	a	b	a	a	b
c	b	a	b	a	a	c
b	b	a	b	a	a	b

π © ρ

a	b	b	a	c	a	b	b	b	b	a	b
b	b	a	a	b	b	b	a	b	a	a	b
c	b	a	a	a	c	b	a	b	a	a	c
b	b	a	a	a	b	b	a	b	a	a	b

π ® ρ

ANEPP


Output node
π
Input node
RSubstitution: a → b(-)
a → c(-)
CDeletion: a → λ (|)
b → λ (|)
RDeletion: a → λ (-)
CSubstitution: a → b (|)
c → d (|)
Output
CDeletion: a → (|)

FORMAL DEFINITIONS (1)

EVOLUTIONARY OPERATIONS AND ACTIONS:

CSubstitution: a → b (|)

$\sigma^*(\pi)$

a	b	b	a	c
b	b	a	a	b
c	b	a	a	a
b	b	a	a	a

a	b	b	a	c
b	b	a	a	b
c	b	a	b	a
b	b	a	a	a

b	b	b	a	c
b	b	a	a	b
c	b	a	a	a
b	b	a	a	a

FORMAL DEFINITIONS (2)

EVOLUTIONARY OPERATIONS AND ACTIONS:

CSubstitution: a → b (|) $\sigma^{\leftarrow}(\pi)$

a	b	b	a	c
b	b	a	a	b
c	b	a	a	a
b	b	a	a	a

b	b	b	a	c
b	b	a	a	b
c	b	a	a	a
b	b	a	a	a

FORMAL DEFINITIONS (3)

EVOLUTIONARY OPERATIONS AND ACTIONS:

CSubstitution:a $\rightarrow$ b (|) $\sigma^{\rightarrow}(\pi)$

a	b	b	a	c
b	b	a	a	b
c	b	a	a	a
b	b	a	a	a

a	b	b	a	c
b	b	a	a	b
c	b	a	a	b
b	b	a	a	a

a	b	b	a	c
b	b	a	a	b
c	b	a	a	a
b	b	a	a	b

FORMAL DEFINITIONS (4)

EVOLUTIONARY OPERATIONS AND ACTIONS:

RDeletion: a → λ (-)

$\sigma^*(\pi)$

a	b	b	a	c
b	b	a	a	b
c	b	a	a	a
b	b	a	a	a

a	b	b	a	c
b	b	a	a	b
b	b	a	a	a

b	b	a	a	b
c	b	a	a	a
b	b	a	a	a

FORMAL DEFINITIONS (5)

EVOLUTIONARY OPERATIONS AND ACTIONS:

RDeletion: a → λ (-) $\sigma^{\uparrow}(\pi)$

a	b	b	a	c
b	b	a	a	b
c	b	a	a	a
b	b	a	a	a

b	b	a	a	b
c	b	a	a	a
b	b	a	a	a

FORMAL DEFINITIONS (6)

EVOLUTIONARY OPERATIONS AND ACTIONS:

RDeletion:a $\rightarrow$ λ (-) $\sigma^{\downarrow}(\pi)$

a	b	b	a	c
b	b	a	a	b
c	b	a	a	a
b	b	a	a	a

a	b	b	a	c
b	b	a	a	b
c	b	a	a	a

FORMAL DEFINITIONS (7)

FILTERS:

$$\varphi^{(s)}(w;P,F) \equiv P \subseteq alph(w) \quad \wedge \quad F \cap alph(w)=\emptyset.$$

$$\varphi^{(w)}(w;P,F) \equiv alph(w) \cap P \neq \emptyset \quad \wedge \quad F \cap alph(w)=\emptyset.$$

$$\varphi^{\beta}(L,P,F)=\{w \in L : \varphi^{\beta}(w;P,F)\}.$$

FORMAL DEFINITIONS (8)

EVOLUTIONARY PICTURE PROCESSOR: (M, PI, FI, PO, FO)

ANEPP: $\Gamma=(V, U, G, \mathcal{N}, \alpha, \beta, x_I, Out)$

$G=(X_G, E_G)$ **: underlying graph structure**

$\mathcal{N}: X_G \rightarrow EP_V$ **: associated evolutionary picture processors**

$\alpha : X_G \rightarrow \{*, \leftarrow, \rightarrow, \downarrow, \uparrow\}$ **: action mode**

$\beta : X_G \rightarrow \{s, w\}$**: filter type**

$\rho_x(.)=\varphi^{\beta(x)}(.; PI_x, FI_x)$**: input filter**

$\tau_x(.)=\varphi^{\beta(x)}(.; PO_x, FO_x)$**: output filter**

FORMAL DEFINITIONS (9)

WORKING MODE

Evolutionary step:

$$C \Rightarrow C', \text{ iff } C'(x) = M_x^{\alpha}(x)(C(x))$$

Communication step:

C ⏩ C' iff

$$C'(x) = (C(x) - \tau_x(C(x))) \cup \bigcup_{\{x,y\} \in EG} (\tau_y(C(y)) \cap \rho_x(C(y)))$$

WEAKLY ACCEPTED LANGUAGE

1. There exists a configuration in which the set of pictures existing in ANY output node is non-empty.

2. There exist two consecutive identical configurations.

3. **It works forever.**

$L_{wa}(\Gamma)=\{\pi$: the computation of Γ on π is an accepting one$\}$.

L is weakly decided by Γ iff $L_{wa}(\Gamma)= L$ and Γ halts on every input

STRONGLY ACCEPTED LANGUAGE

1. **There exists a configuration in which the set of pictures existing in ALL output node is non-empty.**

2. **There exist two consecutive identical configurations.**

3. **It works forever.**

$$L_{sa}(\Gamma)=\{\pi : \text{the computation of } \Gamma \text{ on } \pi \text{ is an accepting one}\}.$$

L is weakly decided by Γ iff $L_{sa}(\Gamma) = L$ and Γ halts on every input

COMPUTATIONAL POWER (1)

Ex1. *Let L be the set of all pictures π over an alphabet V with two identical rows. The language L can be formally described as*

$$L = \{\pi \mid \pi(1; i) = \pi(2; i);\ i \in [m];\ m \geq 1\}$$

Ex2. *Let L be the set of all pictures π over an alphabet V with two identical rows. The language L can be formally described as*

$$L = \{\pi \mid \pi(i; k) = \pi(j; k);\ k \in [m];\ m \geq 1,\ i,j \in [n];\ n \geq 1\}$$

COMPUTATIONAL POWER (2)

Th1. $L_{wa}(\mathbf{ANEPP}) \subseteq L_{sa}(\mathbf{ANEPP})$.

Th2. *The complement of every local language can be weakly accepted by an ANEPP.*

Th3. $L_{wa}(\mathbf{ANEPP}) \setminus \mathbf{REC} \neq \Phi$

CLOSURE PROPERTIES

Th1. ***L_{wa}*(ANEPP)** *is closed under rotation, boolean union, projection, inverse projection.*

Th2. ***L_{sa}*(ANEPP)** *is closed under rotation, boolean intersection, projection, inverse projection.*

2-DIMENSIONAL PATTERN MATCHING

Problem. Given a pattern π and a picture θ is π a subpicture of θ?

Th1. ***For any picture π of size (i,n), $1 \leq i \leq 3$ and $n \geq 1$, there exists an ANEPP Γ such that $L_{wa}(\Gamma)=\{\pi\}$.***

Th2. ***The 2D pattern matching can be weakly decided for patterns of sizes (i,n) with $1 \leq i \leq 3$ and $n \geq 1$.***

2-DIMENSIONAL PATTERN MATCHING

Th1. ***For any picture π of size (i,n), with i, n $\geq$ 1, there exists an ANEPP Γ such that $L_{sa}(\Gamma)=\{\pi\}$.***

OPEN. ***The 2D pattern matching can be strongly decided for arbitrary patterns?***

Thank You