Properties of Eco-Colonies

Šárka Vavrečková, Alica Kelemenová

Institute of Computer Science, Faculty of Philosophy and Science
Silesian University in Opava, Czech Republic

{sarka.vavreckova,alica.kelemenova}@fpf.slu.cz

ACMC'07, August 30, 2007

Outline

Colonies

About colonies

- Introduced in KELEMEN, J., KELEMENOVÁ, A.: A Grammar-theoretic Treatment of Multiagent Systems. *Cybernetics and Systems 23*, 1992, pp. 621–633,
- as collections of simple grammars (components) working on common environment, in relation to multiagent systems,
- sequential, sequential with parallely working components and parallel basic variants,
- parallel variants studied in DASSOW, J., KELEMEN, J., PĂUN, G.: On Parallelism in Colonies. *Cybernetics and Systems 24*, 1993, pp. 37–49.

Colonies

About colonies

- Introduced in Kelemen, J., Kelemenová, A.: A Grammar-theoretic Treatment of Multiagent Systems. *Cybernetics and Systems 23*, 1992, pp. 621–633,
- as collections of simple grammars (components) working on common environment, in relation to multiagent systems,
- sequential, sequential with parallely working components and parallel basic variants,
- parallel variants studied in Dassow, J., Kelemen, J., Păun, G.: On Parallelism in Colonies. *Cybernetics and Systems 24*, 1993, pp. 37–49.

Colonies

About colonies

- Introduced in KELEMEN, J., KELEMENOVÁ, A.: A Grammar-theoretic Treatment of Multiagent Systems. *Cybernetics and Systems 23*, 1992, pp. 621–633,
- as collections of simple grammars (components) working on common environment, in relation to multiagent systems,
- sequential, sequential with parallely working components and parallel basic variants,
- parallel variants studied in DASSOW, J., KELEMEN, J., PĂUN, G.: On Parallelism in Colonies. *Cybernetics and Systems 24*, 1993, pp. 37–49.

Colonies

About colonies

- Introduced in KELEMEN, J., KELEMENOVÁ, A.: A Grammar-theoretic Treatment of Multiagent Systems. *Cybernetics and Systems 23*, 1992, pp. 621–633,
- as collections of simple grammars (components) working on common environment, in relation to multiagent systems,
- sequential, sequential with parallely working components and parallel basic variants,
- parallel variants studied in DASSOW, J., KELEMEN, J., PĂUN, G.: On Parallelism in Colonies. *Cybernetics and Systems 24*, 1993, pp. 37–49.

Eco-grammar systems

About eco-grammar systems

- Introduced in CSUHAJ-VARJÚ, E., KELEMEN, J., KELEMENOVÁ, A., PĂUN, G.: Eco-grammar Systems. Grammatical Framework for Studying Lifelike Interactions. *Artificial Life 3*, 1997, pp. 1–28,
- ecosystem where agents influence state of environment and environment influences states of agents,
- environment is 0L-scheme, all active agents work in each step.

Eco-grammar systems

About eco-grammar systems

- Introduced in Csuhaj-Varjú, E., Kelemen, J., Kelemenová, A., Păun, G.: Eco-grammar Systems. Grammatical Framework for Studying Lifelike Interactions. *Artificial Life 3*, 1997, pp. 1–28,
- ecosystem where agents influence state of environment and environment influences states of agents,
- environment is 0L-scheme, all active agents work in each step.

Eco-grammar systems

About eco-grammar systems

- Introduced in CSUHAJ-VARJÚ, E., KELEMEN, J., KELEMENOVÁ, A., PĂUN, G.: Eco-grammar Systems. Grammatical Framework for Studying Lifelike Interactions. *Artificial Life 3*, 1997, pp. 1–28,
- ecosystem where agents influence state of environment and environment influences states of agents,
- environment is 0L-scheme, all active agents work in each step.

Origin of eco-colonies

About eco-colonies

- Based on colonies with inspiration on eco-grammar systems,
- eco-colonies:
  - simple grammars (agents, components) – as in colonies,
  - self-developing environment – as in eco-grammar systems,
- two types:
  - 0L eco-colonies (environment is 0L-scheme) – as in eco-grammar systems,
  - E0L eco-colonies (environment is E0L-scheme) – as in colonies.
- similar: e-colonies (environment is T0L-scheme) in CSUHAJ-VARJÚ, E.: Colonies – A Multi-agent Approach to Language Generation. In: *Proc. ECAI'96 Workshop on Finite State Models of Languages*, NJSZT, Budapest, 1996, pp. 12–16

Origin of eco-colonies

About eco-colonies

- Based on colonies with inspiration on eco-grammar systems,
- eco-colonies:
  - simple grammars (agents, components) – as in colonies,
  - self-developing environment – as in eco-grammar systems,
- two types:
  - 0L eco-colonies (environment is 0L-scheme) – as in eco-grammar systems,
  - E0L eco-colonies (environment is E0L-scheme) – as in colonies.
- similar: e-colonies (environment is T0L-scheme) in CSUHAJ-VARJÚ, E.: Colonies – A Multi-agent Approach to Language Generation. In: *Proc. ECAI'96 Workshop on Finite State Models of Languages*, NJSZT, Budapest, 1996, pp. 12–16

Origin of eco-colonies

About eco-colonies

- Based on colonies with inspiration on eco-grammar systems,
- eco-colonies:
  - simple grammars (agents, components) – as in colonies,
  - self-developing environment – as in eco-grammar systems,
- two types:
  - 0L eco-colonies (environment is 0L-scheme) – as in eco-grammar systems,
  - E0L eco-colonies (environment is E0L-scheme) – as in colonies.
- similar: e-colonies (environment is T0L-scheme) in CSUHAJ-VARJÚ, E.: Colonies – A Multi-agent Approach to Language Generation. In: *Proc. ECAI'96 Workshop on Finite State Models of Languages*, NJSZT, Budapest, 1996, pp. 12–16

Origin of eco-colonies

About eco-colonies

- Based on colonies with inspiration on eco-grammar systems,
- eco-colonies:
  - simple grammars (agents, components) – as in colonies,
  - self-developing environment – as in eco-grammar systems,
- two types:
  - 0L eco-colonies (environment is 0L-scheme) – as in eco-grammar systems,
  - E0L eco-colonies (environment is E0L-scheme) – as in colonies.
- similar: e-colonies (environment is T0L-scheme) in CSUHAJ-VARJÚ, E.: Colonies – A Multi-agent Approach to Language Generation. In: *Proc. ECAI'96 Workshop on Finite State Models of Languages*, NJSZT, Budapest, 1996, pp. 12–16

Eco-colony as a grammar system

= model of a community of cooperating processes, grammar system

- *symbols* – elements of the alphabet, objects,
- *environment* – contains symbols, the environment is self-developing,
- *word* – some of states of the environment,
- *agents (components)* – cooperating grammars, processes, subjects, working parallely,
  - *start symbol* – what the agent can process, it looks for this symbol in the environment,
  - *finite language of the agent* (set of actions) – what the agent can do with its start symbol, the agent replaces it by some word of this language.

Eco-colony as a grammar system

= model of a community of cooperating processes, grammar system

- *symbols* – elements of the alphabet, objects,
- *environment* – contains symbols, the environment is self-developing,
- *word* – some of states of the environment,
- *agents (components)* – cooperating grammars, processes, subjects, working parallely,
  - *start symbol* – what the agent can process, it looks for this symbol in the environment,
  - *finite language of the agent* (set of actions) – what the agent can do with its start symbol, the agent replaces it by some word of this language.

Eco-colony as a grammar system

= model of a community of cooperating processes, grammar system

- *symbols* – elements of the alphabet, objects,
- *environment* – contains symbols, the environment is self-developing,
- *word* – some of states of the environment,
- *agents (components)* – cooperating grammars, processes, subjects, working parallely,
  - *start symbol* – what the agent can process, it looks for this symbol in the environment,
  - *finite language of the agent* (set of actions) – what the agent can do with its start symbol, the agent replaces it by some word of this language.

Eco-colony as a grammar system

= model of a community of cooperating processes, grammar system

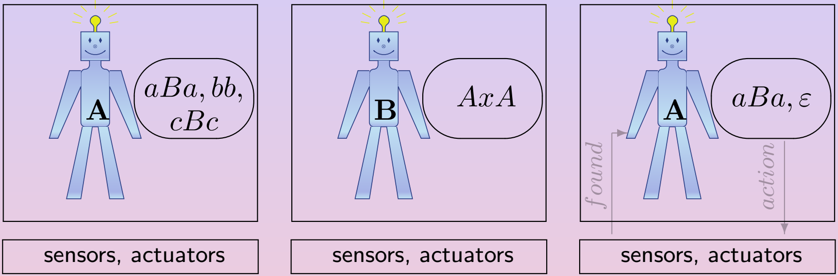
- *symbols* – elements of the alphabet, objects,
- *environment* – contains symbols, the environment is self-developing,
- *word* – some of states of the environment,
- *agents (components)* – cooperating grammars, processes, subjects, working parallely,
  - *start symbol* – what the agent can process, it looks for this symbol in the environment,
  - *finite language of the agent* (set of actions) – what the agent can do with its start symbol, the agent replaces it by some word of this language.

Eco-colony as a grammar system

= model of a community of cooperating processes, grammar system

- *symbols* – elements of the alphabet, objects,
- *environment* – contains symbols, the environment is self-developing,
- *word* – some of states of the environment,
- *agents (components)* – cooperating grammars, processes, subjects, working parallely,
  - *start symbol* – what the agent can process, it looks for this symbol in the environment,
  - *finite language of the agent* (set of actions) – what the agent can do with its start symbol, the agent replaces it by some word of this language.

Example (motivation)



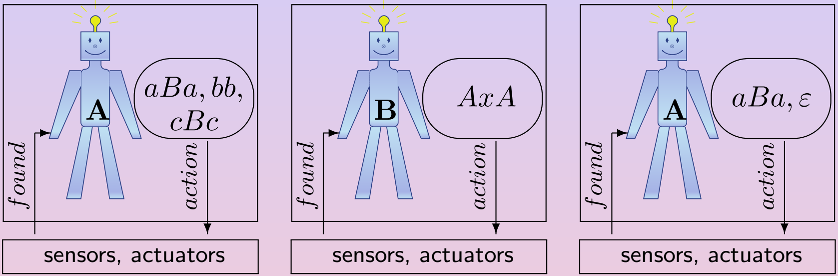
a	A	x	A	a	d	C	C	b								

$B \to bB$ $\quad A \to A$ $\quad C \to c$ $\quad C \to cD$ $\quad d \to d$

$x \to x$ $\quad c \to c$ $\quad b \to bb$

$a \to a$ $\quad D \to C$

Example (motivation)



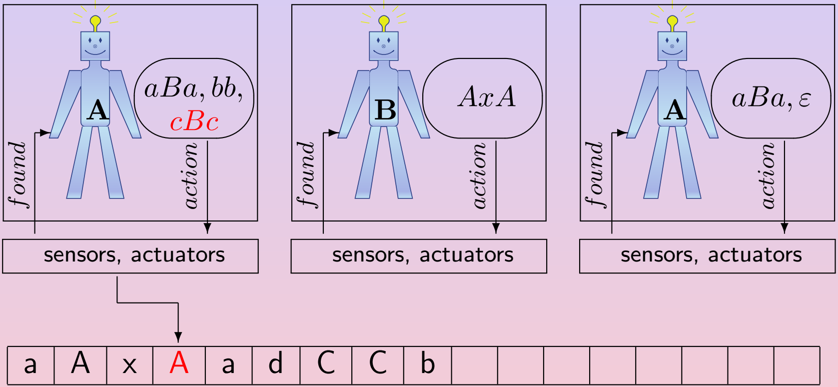
a	A	x	A	a	d	C	C	b								

$B \to bB$ $\quad A \to A$ $\quad C \to c$ $\quad d \to d$

$c \to c$ $\quad C \to cD$ $\quad b \to bb$

$x \to x$ $\quad a \to a$ $\quad D \to C$

Example (motivation)

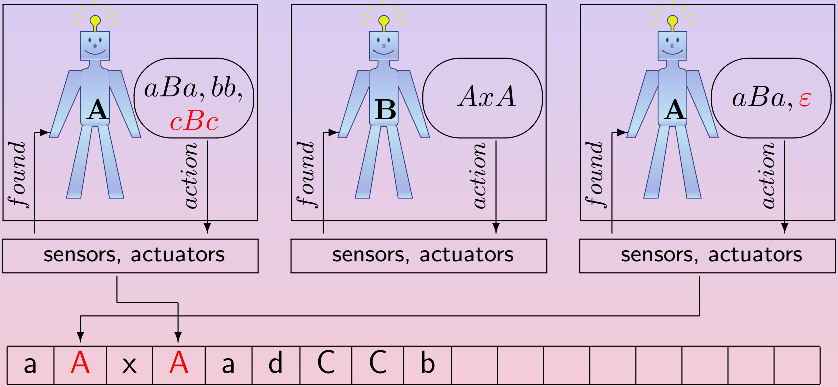


$B \rightarrow bB$ $\quad A \rightarrow A$ $\quad C \rightarrow c$ $\quad C \rightarrow cD$ $\quad d \rightarrow d$

$x \rightarrow x$ $\quad c \rightarrow c$ $\quad b \rightarrow bb$

$a \rightarrow a$ $\quad D \rightarrow C$

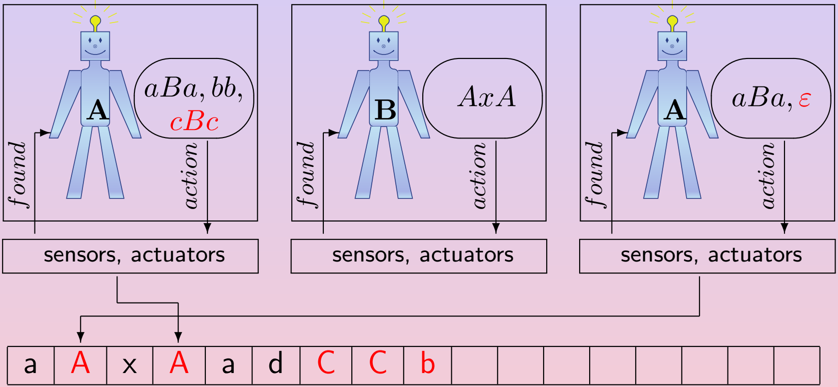
Example (motivation)



$$B \to bB \quad A \to A \quad C \to c \quad c \to c \quad C \to cD \quad d \to d \quad b \to bb$$

$$x \to x \quad a \to a \quad D \to C$$

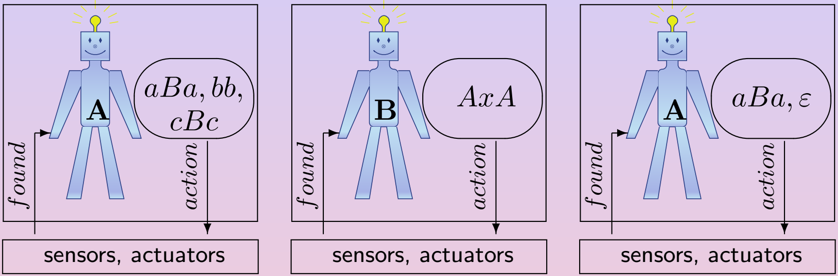
Example (motivation)



$B \to bB \quad A \to A \quad C \to c \quad c \to c \quad C \to cD \quad d \to d \quad b \to bb$

$x \to x \quad a \to a \quad D \to C$

Example (motivation)



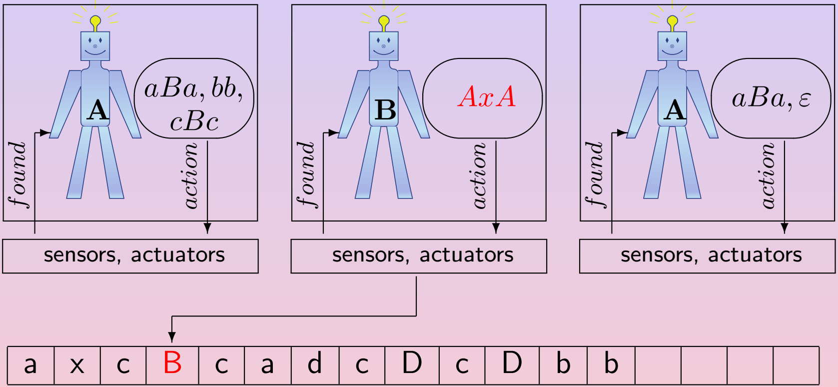
a	x	c	B	c	a	d	c	D	c	D	b	b				

$B \rightarrow bB$ $A \rightarrow A$ $C \rightarrow c$ $d \rightarrow d$

$c \rightarrow c$ $C \rightarrow cD$ $b \rightarrow bb$

$x \rightarrow x$ $a \rightarrow a$ $D \rightarrow C$

Example (motivation)

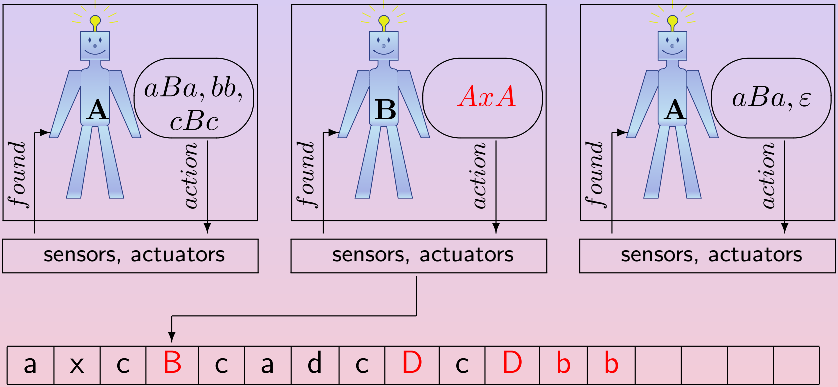


$B \rightarrow bB$ $A \rightarrow A$ $C \rightarrow c$ $C \rightarrow cD$ $d \rightarrow d$

$x \rightarrow x$ $c \rightarrow c$ $b \rightarrow bb$

$a \rightarrow a$ $D \rightarrow C$

Example (motivation)

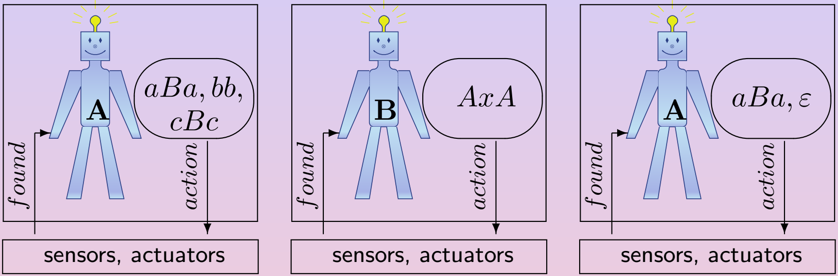


$B \to bB$ $A \to A$ $C \to c$ $C \to cD$ $d \to d$

$c \to c$ $b \to bb$

$x \to x$ $a \to a$ $D \to C$

Example (motivation)



a	x	c	A	x	A	c	a	d	c	C	c	C	b	b	b	b

$B \to bB \quad A \to A \quad x \to x$

$C \to c \quad c \to c \quad C \to cD \quad a \to a$

$d \to d \quad b \to bb \quad D \to C$

Example (more formally)

Agents:

$A_1 = (A,\ \{aBa,\ bb,\ cBc\})$
$A_2 = (B,\ \{AxA\})$
$A_3 = (A,\ \{aBa,\ \varepsilon\})$

Rules in the environment:

$C \to cD$	$D \to C$	$A \to A$	$d \to d$
$C \to c$	$B \to bB$	$a \to a$	$x \to x$
$b \to bb$		$c \to c$	

Alphabets:

Alphabet of the environment: $V = \{A, B, C, D, a, b, c, d, x\}$
Terminal alphabet: $T = \{a, b, c, d, x\}$

Example (more formally)

Agents:

$A_1 = (A,\ \{aBa,\ bb,\ cBc\})$
$A_2 = (B,\ \{AxA\})$
$A_3 = (A,\ \{aBa,\ \varepsilon\})$

Rules in the environment:

$C \rightarrow cD$	$D \rightarrow C$	$A \rightarrow A$	$d \rightarrow d$
$C \rightarrow c$	$B \rightarrow bB$	$a \rightarrow a$	$x \rightarrow x$
$b \rightarrow bb$		$c \rightarrow c$	

Alphabets:

Alphabet of the environment: $V = \{A, B, C, D, a, b, c, d, x\}$
Terminal alphabet: $T = \{a, b, c, d, x\}$

Example (more formally)

Agents:

$A_1 = (A,\ \{aBa,\ bb,\ cBc\})$
$A_2 = (B,\ \{AxA\})$
$A_3 = (A,\ \{aBa,\ \varepsilon\})$

Rules in the environment:

$C \rightarrow cD$	$D \rightarrow C$	$A \rightarrow A$	$d \rightarrow d$
$C \rightarrow c$	$B \rightarrow bB$	$a \rightarrow a$	$x \rightarrow x$
$b \rightarrow bb$		$c \rightarrow c$	

Alphabets:

Alphabet of the environment: $V = \{A, B, C, D, a, b, c, d, x\}$
Terminal alphabet: $T = \{a, b, c, d, x\}$

Definition (Colony)

Colony is a (n+3)-tuple $\mathcal{C} = (V, T, A_1, \ldots A_n, w_0)$, where

- V is a total (finite and non-empty) alphabet of the colony,
- T is a non-empty terminal alphabet of the colony, $T \subset V$,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is a component, where
  - $S_i \in V$ is the start symbol of the component,
  - $F_i \subseteq (V - \{S_i\})^*$, F_i is the finite language of this component,
- w_0 is the axiom.

Definition (Colony)

Colony is a (n+3)-tuple $\mathcal{C} = (V, T, A_1, \ldots A_n, w_0)$, where

- V is a total (finite and non-empty) alphabet of the colony,
- T is a non-empty terminal alphabet of the colony, $T \subset V$,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is a component, where
  - $S_i \in V$ is the start symbol of the component,
  - $F_i \subseteq (V - \{S_i\})^*$, F_i is the finite language of this component,
- w_0 is the axiom.

Definition (Colony)

Colony is a (n+3)-tuple $\mathcal{C} = (V, T, A_1, \ldots A_n, w_0)$, where

- V is a total (finite and non-empty) alphabet of the colony,
- T is a non-empty terminal alphabet of the colony, $T \subset V$,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is a component, where
  - $S_i \in V$ is the start symbol of the component,
  - $F_i \subseteq (V - \{S_i\})^*$, F_i is the finite language of this component,
- w_0 is the axiom.

Definition (Colony)

Colony is a (n+3)-tuple $\mathcal{C} = (V, T, A_1, \ldots A_n, w_0)$, where

- V is a total (finite and non-empty) alphabet of the colony,
- T is a non-empty terminal alphabet of the colony, $T \subset V$,
- $A_i = (S_i, F_i),\ 1 \leq i \leq n$, is a component, where
  - $S_i \in V$ is the start symbol of the component,
  - $F_i \subseteq (V - \{S_i\})^*$, F_i is the finite language of this component,
- w_0 is the axiom.

Definition (Colony)

Colony is a (n+3)-tuple $\mathcal{C} = (V, T, A_1, \ldots A_n, w_0)$, where

- V is a total (finite and non-empty) alphabet of the colony,
- T is a non-empty terminal alphabet of the colony, $T \subset V$,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is a component, where
  - $S_i \in V$ is the start symbol of the component,
  - $F_i \subseteq (V - \{S_i\})^*$, F_i is the finite language of this component,
- w_0 is the axiom.

Definition (Colony)

Colony is a (n+3)-tuple $\mathcal{C} = (V, T, A_1, \ldots A_n, w_0)$, where

- V is a total (finite and non-empty) alphabet of the colony,
- T is a non-empty terminal alphabet of the colony, $T \subset V$,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is a component, where
    - $S_i \in V$ is the start symbol of the component,
    - $F_i \subseteq (V - \{S_i\})^*$, F_i is the finite language of this component,
- w_0 is the axiom.

Derivation steps in colonies

- b-*mode* – *sequential* type of derivation, one component is active in one derivation step, the active component replaces one occurrence of its start symbol by some word of F,
- t-*mode* – *sequentially-parallel* – one component is active in one derivation step, rewriting all occurrences of its start symbol,
- wp-*mode* – *weakly parallel* mode, components which can work must work, each component rewrites at most one occurrence of its start symbol not occupied by another component,
- sp-*mode* – *strongly parallel* mode similar to wp, but if there is an occurrence of a symbol in the environment, every component with this start symbol has to be active – if all occurrences of this symbol are occupied by another components, the derivation is *blocked*.

Derivation steps in colonies

- *b-mode* – *sequential* type of derivation, one component is active in one derivation step, the active component replaces one occurrence of its start symbol by some word of F,
- *t-mode* – *sequentially-parallel* – one component is active in one derivation step, rewriting all occurrences of its start symbol,
- *wp-mode* – *weakly parallel* mode, components which can work must work, each component rewrites at most one occurrence of its start symbol not occupied by another component,
- *sp-mode* – *strongly parallel* mode similar to wp, but if there is an occurrence of a symbol in the environment, every component with this start symbol has to be active – if all occurrences of this symbol are occupied by another components, the derivation is *blocked*.

Derivation steps in colonies

- b*-mode* – *sequential* type of derivation, one component is active in one derivation step, the active component replaces one occurrence of its start symbol by some word of F,
- t*-mode* – *sequentially-parallel* – one component is active in one derivation step, rewriting all occurrences of its start symbol,
- wp*-mode* – *weakly parallel* mode, components which can work must work, each component rewrites at most one occurrence of its start symbol not occupied by another component,
- sp*-mode* – *strongly parallel* mode similar to wp, but if there is an occurrence of a symbol in the environment, every component with this start symbol has to be active – if all occurrences of this symbol are occupied by another components, the derivation is *blocked*.

Derivation steps in colonies

- b-*mode* – *sequential* type of derivation, one component is active in one derivation step, the active component replaces one occurrence of its start symbol by some word of F,
- t-*mode* – *sequentially-parallel* – one component is active in one derivation step, rewriting all occurrences of its start symbol,
- wp-*mode* – *weakly parallel* mode, components which can work must work, each component rewrites at most one occurrence of its start symbol not occupied by another component,
- sp-*mode* – *strongly parallel* mode similar to wp, but if there is an occurrence of a symbol in the environment, every component with this start symbol has to be active – if all occurrences of this symbol are occupied by another components, the derivation is *blocked*.

Definition (Language of colony)

Let $\mathcal{C}$ be a colony and $\mathcal{C} = (V, T, A_1, \ldots A_n, w_0)$. The language generated by the derivation step x, $x \in \{b, t, wp, sp\}$ in $\mathcal{C}$ is

$$L(\mathcal{C}, x) = \left\{ w \in T^* \; : \; w_0 \stackrel{x}{\Longrightarrow}^{*} w \right\}$$

Definition (E0L eco-colony)

An E0L eco-colony of degree n, $n \geq 1$, is an $(n+2)$-tuple $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$, where

- $E = (V, T, P)$ is E0L scheme, where
  - V is an alphabet,
  - T is a terminal alphabet, $T \subseteq V$,
  - P is a finite set of E0L rewriting rules over V,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is the i-th agent, where
  - $S_i \in V$ is the start symbol of the agent,
  - $F_i \subseteq (V - \{S_i\})^*$ is a finite set of action rules of the agent (the language of the agent),
- w_0 is the axiom.

Definition (E0L eco-colony)

An E0L eco-colony of degree n, $n \geq 1$, is an $(n+2)$-tuple $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$, where

- $E = (V, T, P)$ is E0L scheme, where
  - V is an alphabet,
  - T is a terminal alphabet, $T \subseteq V$,
  - P is a finite set of E0L rewriting rules over V,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is the i-th agent, where
  - $S_i \in V$ is the start symbol of the agent,
  - $F_i \subseteq (V - \{S_i\})^*$ is a finite set of action rules of the agent (the language of the agent),
- w_0 is the axiom.

Definition (E0L eco-colony)

An E0L eco-colony of degree n, $n \geq 1$, is an $(n+2)$-tuple $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$, where

- $E = (V, T, P)$ is E0L scheme, where
  - V is an alphabet,
  - T is a terminal alphabet, $T \subseteq V$,
  - P is a finite set of E0L rewriting rules over V,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is the i-th agent, where
  - $S_i \in V$ is the start symbol of the agent,
  - $F_i \subseteq (V - \{S_i\})^*$ is a finite set of action rules of the agent (the language of the agent),
- w_0 is the axiom.

Definition (E0L eco-colony)

An E0L eco-colony of degree n, $n \geq 1$, is an $(n+2)$-tuple $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$, where

- $E = (V, T, P)$ is E0L scheme, where
  - V is an alphabet,
  - T is a terminal alphabet, $T \subseteq V$,
  - P is a finite set of E0L rewriting rules over V,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is the i-th agent, where
  - $S_i \in V$ is the start symbol of the agent,
  - $F_i \subseteq (V - \{S_i\})^*$ is a finite set of action rules of the agent (the language of the agent),
- w_0 is the axiom.

Definition (E0L eco-colony)

An E0L eco-colony of degree n, $n \geq 1$, is an $(n+2)$-tuple $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$, where

- $E = (V, T, P)$ is E0L scheme, where
  - V is an alphabet,
  - T is a terminal alphabet, $T \subseteq V$,
  - P is a finite set of E0L rewriting rules over V,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is the i-th agent, where
  - $S_i \in V$ is the start symbol of the agent,
  - $F_i \subseteq (V - \{S_i\})^*$ is a finite set of action rules of the agent (the language of the agent),
- w_0 is the axiom.

Definition (E0L eco-colony)

An E0L eco-colony of degree n, $n \geq 1$, is an $(n+2)$-tuple $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$, where

- $E = (V, T, P)$ is E0L scheme, where
  - V is an alphabet,
  - T is a terminal alphabet, $T \subseteq V$,
  - P is a finite set of E0L rewriting rules over V,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is the i-th agent, where
  - $S_i \in V$ is the start symbol of the agent,
  - $F_i \subseteq (V - \{S_i\})^*$ is a finite set of action rules of the agent (the language of the agent),
- w_0 is the axiom.

Definition (E0L eco-colony)

An E0L eco-colony of degree n, $n \geq 1$, is an $(n+2)$-tuple $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$, where

- $E = (V, T, P)$ is E0L scheme, where
  - V is an alphabet,
  - T is a terminal alphabet, $T \subseteq V$,
  - P is a finite set of E0L rewriting rules over V,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is the i-th agent, where
  - $S_i \in V$ is the start symbol of the agent,
  - $F_i \subseteq (V - \{S_i\})^*$ is a finite set of action rules of the agent (the language of the agent),
- w_0 is the axiom.

Definition (E0L eco-colony)

An E0L eco-colony of degree n, $n \geq 1$, is an $(n+2)$-tuple $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$, where

- $E = (V, T, P)$ is E0L scheme, where
  - V is an alphabet,
  - T is a terminal alphabet, $T \subseteq V$,
  - P is a finite set of E0L rewriting rules over V,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is the i-th agent, where
  - $S_i \in V$ is the start symbol of the agent,
  - $F_i \subseteq (V - \{S_i\})^*$ is a finite set of action rules of the agent (the language of the agent),
- w_0 is the axiom.

Definition (0L eco-colony)

An 0L eco-colony of degree n, $n \geq 1$, is an $(n+2)$-tuple $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$, where

- $E = (V, P)$ is 0L scheme, where
  - V is an alphabet,
  - (T is not used),
  - P is a finite set of 0L rewriting rules over V,
- $A_i = (S_i, F_i)$, $1 \leq i \leq n$, is the i-th agent, where
  - $S_i \in V$ is the start symbol of the agent,
  - $F_i \subseteq (V - \{S_i\})^*$ is a finite set of action rules of the agent (the language of the agent),
- w_0 is the axiom.

Definition (Weakly competitive parallel derivation step wp)

We define a weakly competitive parallel derivation step in an eco-colony $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$ as the relation $\overset{wp}{\Longrightarrow}$ – α directly derives β in wp mode of derivation (written as $\alpha \overset{wp}{\Longrightarrow} \beta$) if

- $\alpha = v_0 S_{i_1} v_1 S_{i_2} v_2 \ldots v_{r-1} S_{i_r} v_r,\ r > 0,$
- $\beta = v'_0 f_{i_1} v'_1 f_{i_2} v'_2 \ldots v'_{r-1} f_{i_r} v'_r,$ for $A_{i_k} = (S_{i_k}, F_{i_k}),\ f_{i_k} \in F_{i_k},\ 1 \leq k \leq r,$
- $i_k \neq i_m$ for every $k \neq m,\ 1 \leq k, m \leq r$ (the agent A_{i_k} is *active* in this derivation step),
- $\{i_1, i_2, \ldots, i_r\} \subseteq \{1, 2, \ldots, n\},$

Slide 1 of definition

Definition (Weakly competitive parallel derivation step wp)

We define a weakly competitive parallel derivation step in an eco-colony $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$ as the relation $\stackrel{wp}{\Longrightarrow}$ – α directly derives β in wp mode of derivation (written as $\alpha \stackrel{wp}{\Longrightarrow} \beta$) if

- $\alpha = v_0 S_{i_1} v_1 S_{i_2} v_2 \ldots v_{r-1} S_{i_r} v_r, \; r > 0,$
- $\beta = v'_0 f_{i_1} v'_1 f_{i_2} v'_2 \ldots v'_{r-1} f_{i_r} v'_r,$ for $A_{i_k} = (S_{i_k}, F_{i_k}), \; f_{i_k} \in F_{i_k}, \; 1 \leq k \leq r,$
- $i_k \neq i_m$ for every $k \neq m, \; 1 \leq k, m \leq r$ (the agent A_{i_k} is *active* in this derivation step),
- $\{i_1, i_2, \ldots, i_r\} \subseteq \{1, 2, \ldots, n\},$

Definition (Weakly competitive parallel derivation step wp)

We define a weakly competitive parallel derivation step in an eco-colony $\Sigma = (E, A_1, A_2, \dots, A_n, w_0)$ as the relation $\overset{wp}{\Longrightarrow}$ – α directly derives β in wp mode of derivation (written as $\alpha \overset{wp}{\Longrightarrow} \beta$) if

- $\alpha = v_0 S_{i_1} v_1 S_{i_2} v_2 \dots v_{r-1} S_{i_r} v_r, \ r > 0$,
- $\beta = v'_0 f_{i_1} v'_1 f_{i_2} v'_2 \dots v'_{r-1} f_{i_r} v'_r$, for $A_{i_k} = (S_{i_k}, F_{i_k}), \ f_{i_k} \in F_{i_k}, \ 1 \le k \le r$,
- $i_k \neq i_m$ for every $k \neq m, \ 1 \le k, m \le r$ (the agent A_{i_k} is *active* in this derivation step),
- $\{i_1, i_2, \dots, i_r\} \subseteq \{1, 2, \dots, n\}$,

SLIDE 1 OF DEFINITION

Definition (Weakly competitive parallel derivation step wp)

We define a weakly competitive parallel derivation step in an eco-colony $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$ as the relation $\stackrel{wp}{\Longrightarrow}$ – α directly derives β in wp mode of derivation (written as $\alpha \stackrel{wp}{\Longrightarrow} \beta$) if

- $\alpha = v_0 S_{i_1} v_1 S_{i_2} v_2 \ldots v_{r-1} S_{i_r} v_r,\ r > 0,$
- $\beta = v'_0 f_{i_1} v'_1 f_{i_2} v'_2 \ldots v'_{r-1} f_{i_r} v'_r,$ for $A_{i_k} = (S_{i_k}, F_{i_k}),\ f_{i_k} \in F_{i_k},\ 1 \leq k \leq r,$
- $i_k \neq i_m$ for every $k \neq m,\ 1 \leq k, m \leq r$ (the agent A_{i_k} is *active* in this derivation step),
- $\{i_1, i_2, \ldots, i_r\} \subseteq \{1, 2, \ldots, n\},$

Slide 1 of definition

Definition (Weakly competitive parallel derivation step wp)

- for every $S \in V$, if the number of agents with the start symbol S is denoted by t, then

$$\sum_{\substack{j=1 \\ S_{i_j}=S}}^{r} |\alpha|_{S_{i_j}} = min\left(|\alpha|_S, t\right)$$

- $v_k \overset{E}{\Longrightarrow} v'_k,\ v_k \in V^*,\ 0 \leq k \leq r$, is the derivation step of the scheme E.

Slide 2 of definition

Definition (Weakly competitive parallel derivation step wp)

- for every $S \in V$, if the number of agents with the start symbol S is denoted by t, then

$$\sum_{\substack{j=1 \\ S_{i_j}=S}}^{r} |\alpha|_{S_{i_j}} = min\left(|\alpha|_S, t\right)$$

- $v_k \overset{E}{\Longrightarrow} v_k',\ v_k \in V^*,\ 0 \leq k \leq r$, is the derivation step of the scheme E.

Slide 2 of definition

Definition (Derivation step ap – all are working parallely)

We define a derivation step ap (all are working parallely) in an eco-colony $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$ as the relation $\overset{ap}{\Longrightarrow}$ – α directly derives β in ap mode of derivation (written as $\alpha \overset{ap}{\Longrightarrow} \beta$) if

- $\alpha = v_0 S_{i_1} v_1 S_{i_2} v_2 \ldots v_{n-1} S_{i_n} v_n$,
- $\beta = v'_0 f_{i_1} v'_1 f_{i_2} v'_2 \ldots v'_{n-1} f_{i_n} v'_n$, for $A_{i_k} = (S_{i_k}, F_{i_k}),\ f_{i_k} \in F_{i_k},\ 1 \leq k \leq n$,
- $\{i_1, i_2, \ldots, i_n\} = \{1, 2, \ldots, n\}$ (every agent works in every derivation step),
- $v_k \overset{E}{\Longrightarrow} v'_k,\ v_k \in V^*,\ 0 \leq k \leq n$, is the derivation step of the scheme E.

Definition (Derivation step ap – all are working parallely)

We define a derivation step ap (all are working parallely) in an eco-colony $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$ as the relation $\overset{ap}{\Longrightarrow}$ – α directly derives β in ap mode of derivation (written as $\alpha \overset{ap}{\Longrightarrow} \beta$) if

- $\alpha = v_0 S_{i_1} v_1 S_{i_2} v_2 \ldots v_{n-1} S_{i_n} v_n$,
- $\beta = v'_0 f_{i_1} v'_1 f_{i_2} v'_2 \ldots v'_{n-1} f_{i_n} v'_n$, for $A_{i_k} = (S_{i_k}, F_{i_k}),\ f_{i_k} \in F_{i_k},\ 1 \leq k \leq n$,
- $\{i_1, i_2, \ldots, i_n\} = \{1, 2, \ldots, n\}$ (every agent works in every derivation step),
- $v_k \overset{E}{\Longrightarrow} v'_k,\ v_k \in V^*,\ 0 \leq k \leq n$, is the derivation step of the scheme E.

Definition (Derivation step ap – all are working parallely)

We define a derivation step ap (all are working parallely) in an eco-colony $\Sigma = (E, A_1, A_2, \dots, A_n, w_0)$ as the relation $\xrightarrow{ap}$ – α directly derives β in ap mode of derivation (written as $\alpha \xrightarrow{ap} \beta$) if

- $\alpha = v_0 S_{i_1} v_1 S_{i_2} v_2 \dots v_{n-1} S_{i_n} v_n$,
- $\beta = v'_0 f_{i_1} v'_1 f_{i_2} v'_2 \dots v'_{n-1} f_{i_n} v'_n$, for $A_{i_k} = (S_{i_k}, F_{i_k}),\ f_{i_k} \in F_{i_k},\ 1 \leq k \leq n$,
- $\{i_1, i_2, \dots, i_n\} = \{1, 2, \dots, n\}$ (every agent works in every derivation step),
- $v_k \xrightarrow{E} v'_k,\ v_k \in V^*,\ 0 \leq k \leq n$, is the derivation step of the scheme E.

Definition (Derivation step ap – all are working parallely)

We define a derivation step ap (all are working parallely) in an eco-colony $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$ as the relation $\stackrel{ap}{\Longrightarrow}$ – α directly derives β in ap mode of derivation (written as $\alpha \stackrel{ap}{\Longrightarrow} \beta$) if

- $\alpha = v_0 S_{i_1} v_1 S_{i_2} v_2 \ldots v_{n-1} S_{i_n} v_n$,
- $\beta = v'_0 f_{i_1} v'_1 f_{i_2} v'_2 \ldots v'_{n-1} f_{i_n} v'_n$, for $A_{i_k} = (S_{i_k}, F_{i_k}),\ f_{i_k} \in F_{i_k},\ 1 \leq k \leq n$,
- $\{i_1, i_2, \ldots, i_n\} = \{1, 2, \ldots, n\}$ (every agent works in every derivation step),
- $v_k \stackrel{E}{\Longrightarrow} v'_k,\ v_k \in V^*,\ 0 \leq k \leq n$, is the derivation step of the scheme E.

Definition (Language of eco-colony)

Let Σ be an 0L eco-colony, $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$. The language generated by the derivation step x, $x \in \{wp, ap\}$, in Σ is

$$L(\Sigma, x) = \{w \in V^* \ : \ w_0 \overset{x\ *}{\Longrightarrow} w\}.$$

Let Σ be an E0L eco-colony, $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$. The language generated by the derivation step x, $x \in \{wp, ap\}$, in Σ is

$$L(\Sigma, x) = \{w \in T^* \ : \ w_0 \overset{x\ *}{\Longrightarrow} w\}.$$

Definition (Language of eco-colony)

Let Σ be an 0L eco-colony, $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$. The language generated by the derivation step x, $x \in \{wp, ap\}$, in Σ is

$$L(\Sigma, x) = \{w \in V^* \;:\; w_0 \stackrel{x}{\Longrightarrow}{}^* w\}.$$

Let Σ be an E0L eco-colony, $\Sigma = (E, A_1, A_2, \ldots, A_n, w_0)$. The language generated by the derivation step x, $x \in \{wp, ap\}$, in Σ is

$$L(\Sigma, x) = \{w \in T^* \;:\; w_0 \stackrel{x}{\Longrightarrow}{}^* w\}.$$

Example of eco-colony

$\Sigma = (E, A_1, A_2, AbB)$ is an E0L eco-colony, where
$E = (\{A, B, a, b\}, \{a, b\}, \{a \to a, b \to bb, A \to A, B \to B\})$,
$A_1 = (A, \{aB, \varepsilon\})$, $A_2 = (B, \{aA, \varepsilon\})$

$AbB \xRightarrow{ap} aBb^2aA \xRightarrow{ap} a^2Ab^4a^2B \xRightarrow{ap} \dots \xRightarrow{ap} a^nAb^{2^n}a^nB \xRightarrow{ap} \xRightarrow{ap} a^nb^{2^{(n+1)}}a^n$

$AbB \xRightarrow{wp} aBb^2aA \xRightarrow{wp} a^2Ab^4a^2B \xRightarrow{wp} a^2b^8a^3A \xRightarrow{wp} a^2b^{16}a^4B \xRightarrow{wp} \xRightarrow{wp} a^2b^{32}a^5A \xRightarrow{wp} \dots$

The generated languages are:

$$L(\Sigma, ap) = \{a^nb^{2^n}a^n \;:\; n \geq 0\}$$
$$L(\Sigma, wp) = \{a^ib^{2^n}a^j \;:\; n \geq 0,\; 0 \leq i, j < n\}$$

Example of eco-colony

$\Sigma = (E, A_1, A_2, AbB)$ is an E0L eco-colony, where
$E = (\{A, B, a, b\}, \{a, b\}, \{a \to a, b \to bb, A \to A, B \to B\})$,
$A_1 = (A, \{aB, \varepsilon\}), \quad A_2 = (B, \{aA, \varepsilon\})$

$$AbB \stackrel{ap}{\Longrightarrow} aBb^2aA \stackrel{ap}{\Longrightarrow} a^2Ab^4a^2B \stackrel{ap}{\Longrightarrow} \ldots \stackrel{ap}{\Longrightarrow} a^nAb^{2^n}a^nB \stackrel{ap}{\Longrightarrow} \stackrel{ap}{\Longrightarrow} a^nb^{2^{(n+1)}}a^n$$

$$AbB \stackrel{wp}{\Longrightarrow} aBb^2aA \stackrel{wp}{\Longrightarrow} a^2Ab^4a^2B \stackrel{wp}{\Longrightarrow} a^2b^8a^3A \stackrel{wp}{\Longrightarrow} a^2b^{16}a^4B \stackrel{wp}{\Longrightarrow} \stackrel{wp}{\Longrightarrow} a^2b^{32}a^5A \stackrel{wp}{\Longrightarrow} \ldots$$

The generated languages are:

$$L(\Sigma, ap) = \{a^nb^{2^n}a^n \;:\; n \geq 0\}$$

$$L(\Sigma, wp) = \{a^ib^{2^n}a^j \;:\; n \geq 0,\; 0 \leq i, j < n\}$$

Example of eco-colony

$\Sigma = (E, A_1, A_2, AbB)$ is an E0L eco-colony, where
$E = (\{A, B, a, b\}, \{a, b\}, \{a \to a, b \to bb, A \to A, B \to B\}),$
$A_1 = (A, \{aB, \varepsilon\}), \quad A_2 = (B, \{aA, \varepsilon\})$

$AbB \overset{ap}{\Longrightarrow} aBb^2aA \overset{ap}{\Longrightarrow} a^2Ab^4a^2B \overset{ap}{\Longrightarrow} \ldots \overset{ap}{\Longrightarrow} a^nAb^{2^n}a^nB \overset{ap}{\Longrightarrow}$
$\overset{ap}{\Longrightarrow} a^nb^{2^{(n+1)}}a^n$

$AbB \overset{wp}{\Longrightarrow} aBb^2aA \overset{wp}{\Longrightarrow} a^2Ab^4a^2B \overset{wp}{\Longrightarrow} a^2b^8a^3A \overset{wp}{\Longrightarrow} a^2b^{16}a^4B \overset{wp}{\Longrightarrow}$
$\overset{wp}{\Longrightarrow} a^2b^{32}a^5A \overset{wp}{\Longrightarrow} \ldots$

The generated languages are:

$$L(\Sigma, ap) = \{a^nb^{2^n}a^n \ : \ n \geq 0\}$$
$$L(\Sigma, wp) = \{a^ib^{2^n}a^j \ : \ n \geq 0, \ 0 \leq i, j < n\}$$

Notation

$0EC_x$	$x \in \{wp, ap\}$ class of 0L eco-colonies with x type of derivation
EEC_x	$x \in \{wp, ap\}$ class of E0L eco-colonies with x type of derivation
COL_x	$x \in \{b, t, wp, sp\}$ class of colonies with the x type of derivation
COL_x^T	$x \in \{b, t, wp, sp\}$ class of colonies with the x type of derivation, $T = V$

Results proved in the paper

Table 1 – Colonies and eco-colonies

	$0EC_{wp}$	$0EC_{ap}$	EEC_{wp}	EEC_{ap}
COL_b	$\between$	$\between$	$\subset$	$\subset$
COL_t	$\between$	$\between$	$\not\supseteq$	$\not\supseteq$
COL_{wp}	$\between$	$\between$	$\subset$	$\not\supseteq$
COL_{sp}	$\between$	$\between$	$\not\supseteq$	$\not\supseteq$

Results proved in the paper

Table 2 – Colonies with $T = V$ and eco-colonies

	$0EC_{wp}$	$0EC_{ap}$	EEC_{wp}	EEC_{ap}
COL_b^T	$\subset$	$\not\supseteq$	$\subset$	$\subset$
COL_t^T	$\not\supseteq$	$\not\supseteq$	$\not\supseteq$	$\not\supseteq$
COL_{wp}^T	$\subset$	$\not\supseteq$	$\subset$	$\not\supseteq$
COL_{sp}^T	$\not\supseteq$	$\not\supseteq$	$\not\supseteq$	$\not\supseteq$

Results proved in the paper

Table 3 – Various types of eco-colonies

	$0EC_{wp}$	$0EC_{ap}$	EEC_{wp}	EEC_{ap}
$0EC_{wp}$	$=$	$\bigcirc\!\!\!\!\bigcirc$	$\subset$	
$0EC_{ap}$	$\bigcirc\!\!\!\!\bigcirc$	$=$		$\subset$
EEC_{wp}	$\supset$		$=$	
EEC_{ap}		$\supset$		$=$

Lemma

$$COL_x^T \subseteq COL_x$$

where $x \in \{b, t, wp, sp\}$.

Lemma (Pumping lemma for parallel colonies)

Let L be an infinite language generated by a colony $\mathcal{C}$ with $x \in \{wp, sp\}$ derivation mode. Then the length set of L contains infinite linearly dependent subsets, i.e.

$$\{a \cdot i + b \ : \ i \geq 0\} \subseteq \{|w| \ : \ w \in L\}$$

for some natural numbers $a, b > 0$.

Proof.

Let $\mathcal{C}$ be a colony, $\mathcal{C} = (V, T, A_1, \dots A_n, w_0)$, with n components:

$$m = max\left\{|u| \ : \ u \in F_i, \ A_i = (S_i, F_i), \ 1 \leq i \leq n\right\}.$$

We choose $w \in L(\mathcal{C}) \ : \ |w| \geq |w_0| \cdot m \cdot n \cdot 2^n$. For every i, in $w_0 \Rightarrow^* w_i \Rightarrow w_{i+1} \Rightarrow^* w$ is $|w_{i+1}| - |w_i| \leq m \cdot n$
We split this derivation:

$$w_0 \Rightarrow^* w_i \Rightarrow^* w_j \Rightarrow^* w$$

where

- $0 < i < j < 2^n$,
- the same components are active in the both derivation steps $w_i \Rightarrow w_{i+1}$, $w_j \Rightarrow w_{j+1}$

Proof.

Let $\mathcal{C}$ be a colony, $\mathcal{C} = (V, T, A_1, \ldots A_n, w_0)$, with n components:

$$m = max\{|u| \ : \ u \in F_i, \ A_i = (S_i, F_i), \ 1 \leq i \leq n\}.$$

We choose $w \in L(\mathcal{C}) \ : \ |w| \geq |w_0| \cdot m \cdot n \cdot 2^n$. For every i, in $w_0 \Rightarrow^* w_i \Rightarrow w_{i+1} \Rightarrow^* w$ is $|w_{i+1}| - |w_i| \leq m \cdot n$
We split this derivation:

$$w_0 \Rightarrow^* w_i \Rightarrow^* w_j \Rightarrow^* w$$

where

- $0 < i < j < 2^n$,
- the same components are active in the both derivation steps $w_i \Rightarrow w_{i+1}$, $w_j \Rightarrow w_{j+1}$

Proof.

Let $\mathcal{C}$ be a colony, $\mathcal{C} = (V, T, A_1, \dots A_n, w_0)$, with n components:

$$m = max\left\{|u| \ : \ u \in F_i, \ A_i = (S_i, F_i), \ 1 \leq i \leq n\right\}.$$

We choose $w \in L(\mathcal{C}) \ : \ |w| \geq |w_0| \cdot m \cdot n \cdot 2^n$. For every i, in $w_0 \Rightarrow^* w_i \Rightarrow w_{i+1} \Rightarrow^* w$ is $|w_{i+1}| - |w_i| \leq m \cdot n$
We split this derivation:

$$w_0 \Rightarrow^* w_i \Rightarrow^* w_j \Rightarrow^* w$$

where

- $0 < i < j < 2^n$,
- the same components are active in the both derivation steps $w_i \Rightarrow w_{i+1}, \, w_j \Rightarrow w_{j+1}$

Proof.

$$w_0 \Rightarrow^* w_i \Rightarrow^* w_j \Rightarrow^* w$$

Denote by

- n_0 number of terminals generated in the subderivation $w_0 \Rightarrow^* w_i$, which are not rewritten in any next derivation step,
- n_i the same for the subderivation $w_i \Rightarrow^* w_j$,
- n_j the same for the subderivation $w_j \Rightarrow^* w$.

Proof.

$$w_0 \Rightarrow^* w_i \Rightarrow^* w_j \Rightarrow^* w$$

Denote by

- n_0 number of terminals generated in the subderivation $w_0 \Rightarrow^* w_i$, which are not rewritten in any next derivation step,
- n_i the same for the subderivation $w_i \Rightarrow^* w_j$,
- n_j the same for the subderivation $w_j \Rightarrow^* w$.

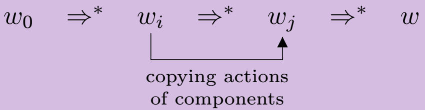
$$w_0 \quad \Rightarrow^* \quad w_i \quad \Rightarrow^* \quad w_j \quad \Rightarrow^* \quad w$$

Proof.

$$w_0 \Rightarrow^* w_i \Rightarrow^* w_j \Rightarrow^* w$$

Denote by

- n_0 number of terminals generated in the subderivation $w_0 \Rightarrow^* w_i$, which are not rewritten in any next derivation step,
- n_i the same for the subderivation $w_i \Rightarrow^* w_j$,
- n_j the same for the subderivation $w_j \Rightarrow^* w$.

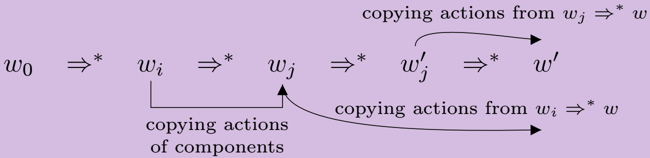


Proof.

$$w_0 \Rightarrow^* w_i \Rightarrow^* w_j \Rightarrow^* w$$

Denote by

- n_0 number of terminals generated in the subderivation $w_0 \Rightarrow^* w_i$, which are not rewritten in any next derivation step,
- n_i the same for the subderivation $w_i \Rightarrow^* w_j$,
- n_j the same for the subderivation $w_j \Rightarrow^* w$.



Proof.

$$w_0 \Rightarrow^* w_i \Rightarrow^* w_j \Rightarrow^* w'_j \Rightarrow^* w'$$

Denote by

- n_0 number of terminals generated in the subderivation $w_0 \Rightarrow^* w_i$, which are not rewritten in any next derivation step,
- n_i the same for the subderivation $w_i \Rightarrow^* w_j$,
- n_j the same for the subderivation $w_j \Rightarrow^* w$.

We do it z-times, $z \geq 0 \implies$ "pumping" the derivation, derived word is w'_z with the length

$$\begin{aligned} |w| = |w'_1| &= n_0 + n_i + n_j \\ |w'| = |w'_2| &= n_0 + 2 \cdot n_i + n_j \\ |w'_z| &= n_0 + z \cdot n_i + n_j \end{aligned}$$

Theorem

$$COL_{wp} \subset EEC_{wp} \tag{1}$$

Proof.

Relation $COL_{wp} \subseteq EEC_{wp}$ is trivial.

Proper subset: $L = \{a^{2^n} \; : \; n \geq 0\} \quad \in \quad EEC_{wp} - COL_{wp}$

- $\in EEC_{wp}$: L is generated by eco-colony $\Sigma = (E, A, b)$, $E = (\{a, b\}, \; \{a\}, \; \{a \to aa, \; b \to b\})$, $A = (b, \; \{a\})$
  $b \overset{wp}{\Longrightarrow} a \overset{wp}{\Longrightarrow} aa \overset{wp}{\Longrightarrow} aaaa \overset{wp}{\Longrightarrow} aaaaaaaa \overset{wp}{\Longrightarrow} \ldots$
- $\notin COL_{wp}$: follows from previous lemma (none $w \in L$ exists such as $w'_z \in L$ for all $z \geq 0$)

□

Theorem

$$COL_{wp} \subset EEC_{wp} \quad (1)$$

Proof.

Relation $COL_{wp} \subseteq EEC_{wp}$ is trivial.

Proper subset: $L = \left\{ a^{2^n} \ : \ n \geq 0 \right\} \quad \in \quad EEC_{wp} - COL_{wp}$

- $\in EEC_{wp}$: L is generated by eco-colony $\Sigma = (E, A, b)$, $E = (\{a, b\},\ \{a\},\ \{a \to aa,\ b \to b\})$, $A = (b,\ \{a\})$
  $b \stackrel{wp}{\Longrightarrow} a \stackrel{wp}{\Longrightarrow} aa \stackrel{wp}{\Longrightarrow} aaaa \stackrel{wp}{\Longrightarrow} aaaaaaaa \stackrel{wp}{\Longrightarrow} \dots$
- $\notin COL_{wp}$: follows from previous lemma (none $w \in L$ exists such as $w'_z \in L$ for all $z \geq 0$)

□

Theorem

$$COL_{wp} \subset EEC_{wp} \tag{1}$$

Proof.

Relation $COL_{wp} \subseteq EEC_{wp}$ is trivial.

Proper subset: $L = \{a^{2^n} \ : \ n \geq 0\} \quad \in \quad EEC_{wp} - COL_{wp}$

- $\in EEC_{wp}$: L is generated by eco-colony $\Sigma = (E, A, b)$, $E = (\{a, b\}, \ \{a\}, \ \{a \to aa, \ b \to b\})$, $A = (b, \ \{a\})$
  $b \overset{wp}{\Longrightarrow} a \overset{wp}{\Longrightarrow} aa \overset{wp}{\Longrightarrow} aaaa \overset{wp}{\Longrightarrow} aaaaaaaa \overset{wp}{\Longrightarrow} \dots$
- $\notin COL_{wp}$: follows from previous lemma (none $w \in L$ exists such as $w'_z \in L$ for all $z \geq 0$)

□

Theorem

$$COL_{wp} \subset EEC_{wp} \tag{1}$$

Proof.

Relation $COL_{wp} \subseteq EEC_{wp}$ is trivial.

Proper subset: $L = \{a^{2^n} \ : \ n \geq 0\} \quad \in \quad EEC_{wp} - COL_{wp}$

- $\in EEC_{wp}$: L is generated by eco-colony $\Sigma = (E, A, b)$, $E = (\{a, b\}, \ \{a\}, \ \{a \to aa, \ b \to b\})$, $A = (b, \ \{a\})$
  $b \overset{wp}{\Longrightarrow} a \overset{wp}{\Longrightarrow} aa \overset{wp}{\Longrightarrow} aaaa \overset{wp}{\Longrightarrow} aaaaaaaa \overset{wp}{\Longrightarrow} \dots$
- $\notin COL_{wp}$: follows from previous lemma (none $w \in L$ exists such as $w'_z \in L$ for all $z \geq 0$)

□

Theorem

$$COL_{wp} \subset EEC_{wp} \tag{1}$$

Proof.

Relation $COL_{wp} \subseteq EEC_{wp}$ is trivial.

Proper subset: $L = \left\{ a^{2^n} \ : \ n \geq 0 \right\} \quad \in \quad EEC_{wp} - COL_{wp}$

- $\in EEC_{wp}$: L is generated by eco-colony $\Sigma = (E, A, b)$, $E = (\{a, b\},\ \{a\},\ \{a \to aa,\ b \to b\})$, $A = (b,\ \{a\})$
  $b \overset{wp}{\Longrightarrow} a \overset{wp}{\Longrightarrow} aa \overset{wp}{\Longrightarrow} aaaa \overset{wp}{\Longrightarrow} aaaaaaaa \overset{wp}{\Longrightarrow} \dots$
- $\notin COL_{wp}$: follows from previous lemma (none $w \in L$ exists such as $w'_z \in L$ for all $z \geq 0$)

□

Theorem

$$COL_b \subset EEC_{ap} \tag{2}$$

The proof can be found in the paper.
We demonstrate the construction of the proof on the colony generating this language:

$$L = \{waw^R a^i \mid w \in \{0,1\}^*,\ i > 0\}$$

Theorem

$$COL_b \subset EEC_{ap} \tag{2}$$

The proof can be found in the paper.
We demonstrate the construction of the proof on the colony generating this language:

$$L = \{waw^R a^i \mid w \in \{0,1\}^*,\ i > 0\}$$

Example: $L = \{waw^R a^i \;:\; w \in \{0,1\}^*,\; i > 0\}$

Colony: $\mathcal{C} = (\{S, H, H', A, A', 0, 1, a\}, \{0, 1, a\}, A_1, \ldots, A_5, S)$
$A_1 = (S, \{HA\})$, $A_2 = (H, \{0H'0, 1H'1, a\})$, $A_4 = (H', \{H\})$,
$A_3 = (A, \{aA', a\})$, $A_5 = (A', \{A\})$ }

Agens work but do not make the word (longer axiom),
components $\Longrightarrow$ rules in the environment.

E0L eco-colony with ap derivation: $\Sigma = (E, A_1, A_2, BCS)$,
$E = (\{B, C, S, H, H', A, A', 0, 1, a\},\ \{0, 1, a\},\ P)$,
$A_1 = (B, \{C, \varepsilon\})$, $A_2 = (C, \{B, \varepsilon\})$,
the set of rules P in the environment is

$$P = \{H \to H|0H'0|1H'1|A,\ H' \to H'|H,\ 1 \to 1,\ a \to a,\ A \to A|aA'|a,\ A' \to A'|A,\ 0 \to 0,\ S \to S|HA\}$$

Example: $L = \{waw^R a^i \ : \ w \in \{0,1\}^*, \ i > 0\}$

Colony: $\mathcal{C} = (\{S, H, H', A, A', 0, 1, a\}, \{0, 1, a\}, A_1, \ldots, A_5, S)$
$A_1 = (S, \{HA\}), \quad A_2 = (H, \{0H'0, 1H'1, a\}), \quad A_4 = (H', \{H\}),$
$A_3 = (A, \{aA', a\}), \quad A_5 = (A', \{A\}) \ \}$

Agens work but do not make the word (longer axiom),
components $\Longrightarrow$ rules in the environment.

E0L eco-colony with ap derivation: $\Sigma = (E, A_1, A_2, BCS)$,
$E = (\{B, C, S, H, H', A, A', 0, 1, a\}, \ \{0, 1, a\}, \ P)$,
$A_1 = (B, \{C, \varepsilon\})$, $A_2 = (C, \{B, \varepsilon\})$,
the set of rules P in the environment is

$$P = \{H \rightarrow H|0H'0|1H'1|A, \ H' \rightarrow H'|H, \ 1 \rightarrow 1, \ a \rightarrow a, \ A \rightarrow A|aA'|a, \ A' \rightarrow A'|A, \ 0 \rightarrow 0, \ S \rightarrow S|HA\}$$

Example: $L = \{waw^R a^i \;:\; w \in \{0,1\}^*,\; i > 0\}$

Colony:

$$S \stackrel{b}{\Rightarrow} HA \stackrel{b}{\Rightarrow} 1H'1A \stackrel{b}{\Rightarrow} 1H1A \stackrel{b}{\Rightarrow} 10H'01A \stackrel{b}{\Rightarrow} \dots$$

E0L eco-colony with ap derivation:

$$BCS \stackrel{ap}{\Longrightarrow} CBHA \stackrel{ap}{\Longrightarrow} BC1H'1A \stackrel{ap}{\Longrightarrow} CB1H1A \stackrel{ap}{\Longrightarrow} BC10H'01A \stackrel{ap}{\Longrightarrow} \dots$$

Example: $L = \{waw^R a^i \ : \ w \in \{0,1\}^*, \ i > 0\}$

Colony:

$$S \overset{b}{\Rightarrow} HA \overset{b}{\Rightarrow} 1H'1A \overset{b}{\Rightarrow} 1H1A \overset{b}{\Rightarrow} 10H'01A \overset{b}{\Rightarrow} \ldots$$

E0L eco-colony with ap derivation:

$$BCS \overset{ap}{\Longrightarrow} CBHA \overset{ap}{\Longrightarrow} BC1H'1A \overset{ap}{\Longrightarrow} CB1H1A \overset{ap}{\Longrightarrow} BC10H'01A \overset{ap}{\Longrightarrow} \ldots$$

Example: $L = \{waw^R a^i \ : \ w \in \{0,1\}^*, \ i > 0\}$

Colony:

$$S \stackrel{b}{\Rightarrow} HA \stackrel{b}{\Rightarrow} 1H'1A \stackrel{b}{\Rightarrow} 1H1A \stackrel{b}{\Rightarrow} 10H'01A \stackrel{b}{\Rightarrow} \dots$$

E0L eco-colony with ap derivation:

$$BCS \stackrel{ap}{\Longrightarrow} CBHA \stackrel{ap}{\Longrightarrow} BC1H'1A \stackrel{ap}{\Longrightarrow} CB1H1A \stackrel{ap}{\Longrightarrow} BC10H'01A \stackrel{ap}{\Longrightarrow} \dots$$

Example: $L = \{waw^R a^i \ : \ w \in \{0,1\}^*, \ i > 0\}$

Colony:

$$S \stackrel{b}{\Rightarrow} HA \stackrel{b}{\Rightarrow} 1H'1A \stackrel{b}{\Rightarrow} 1H1A \stackrel{b}{\Rightarrow} 10H'01A \stackrel{b}{\Rightarrow} \dots$$

E0L eco-colony with ap derivation:

$$BCS \stackrel{ap}{\Longrightarrow} CBHA \stackrel{ap}{\Longrightarrow} BC1H'1A \stackrel{ap}{\Longrightarrow} CB1H1A \stackrel{ap}{\Longrightarrow} BC10H'01A \stackrel{ap}{\Longrightarrow} \dots$$

Example: $L = \{waw^R a^i \ : \ w \in \{0,1\}^*, \ i > 0\}$

Colony:

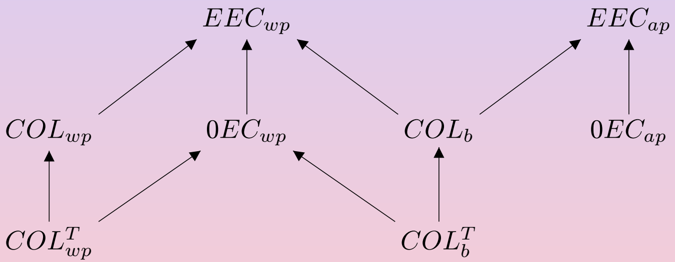
$$S \overset{b}{\Rightarrow} HA \overset{b}{\Rightarrow} 1H'1A \overset{b}{\Rightarrow} 1H1A \overset{b}{\Rightarrow} 10H'01A \overset{b}{\Rightarrow} \dots$$

E0L eco-colony with ap derivation:

$$BCS \overset{ap}{\Longrightarrow} CBHA \overset{ap}{\Longrightarrow} BC1H'1A \overset{ap}{\Longrightarrow} CB1H1A \overset{ap}{\Longrightarrow} BC10H'01A \overset{ap}{\Longrightarrow} \dots$$

Conclusions

Proved proper subsets:



Thank you
for your attention.