Computational models in distributed environments

Katalin Anna Lázár

January 8, 2014
Complex distributed networks
- Formal language theoretic approach
- Networks of language processors
- Simple eco–grammar systems

P2P networks
- $T_{ci}NPMP_{F0L}$ systems
- Information dynamics
- Discretionary Access Control (DAC)

Internet crawlers seeking novel information
- Eco–foraging systems
- The power of eco–foraging systems

Efficiency of goal–oriented Internet crawlers in different graph topologies
- Results

Nature–motivated (bio–inspired) computational models
- Background and motivation
- Research methodology
- Summary and outlook
Complex distributed networks

- a set of units subject to change,
Complex distributed networks

- a set of units subject to change,
- the units are connected to and interact with each other through directed edges or links,
Complex distributed networks

- a set of units subject to change,
- the units are connected to and interact with each other through directed edges or links,
- an alliance is a distinguished set of units,
Complex distributed networks

- a set of units subject to change,
- the units are connected to and interact with each other through directed edges or links,
- an alliance is a distinguished set of units,
- the memory of the alliance is the set of links between the members of the given alliance,
a set of units subject to change,
the units are connected to and interact with each other through directed edges or links,
an alliance is a distinguished set of units,
the memory of the alliance is the set of links between the members of the given alliance,
the work of the alliance is characterized by the rules aiming at the optimization of rewards obtainable by the alliance,
Complex distributed networks

- a set of units subject to change,
- the units are connected to and interact with each other through directed edges or links,
- an alliance is a distinguished set of units,
- the memory of the alliance is the set of links between the members of the given alliance,
- the work of the alliance is characterized by the rules aiming at the optimization of rewards obtainable by the alliance,
- new dependencies may be formed during the work of the alliance on the basis of the interactions between the units and the rewards that the units collect (self-organizing property),
Complex distributed networks

- a set of units subject to change,
- the units are connected to and interact with each other through directed edges or links,
- an alliance is a distinguished set of units,
- the memory of the alliance is the set of links between the members of the given alliance,
- the work of the alliance is characterized by the rules aiming at the optimization of rewards obtainable by the alliance,
- new dependencies may be formed during the work of the alliance on the basis of the interactions between the units and the rewards that the units collect (self-organizing property),
- the reward sharing rules induce competition among the units (selective property).
Grammar systems theory [Csuha–Varjú et al., 1994]:

- The theory of grammar systems regards formal languages as a set of sequences of symbols describing the behaviour of complex systems of cooperating and communicating agents at symbolic level.
Formal language theoretic approach

Grammar systems theory [Csuhaj–Varjú et al., 1994]:

- The theory of grammar systems regards formal languages as a set of sequences of symbols describing the behaviour of complex systems of cooperating and communicating agents at symbolic level.
- The characteristics of a system are determined through the individual and the collective behaviour of its members.
Grammar systems theory [Csuhaj–Varjú et al., 1994]:

- The theory of grammar systems regards formal languages as a set of sequences of symbols describing the behaviour of complex systems of cooperating and communicating agents at symbolic level.
- The characteristics of a system are determined through the individual and the collective behaviour of its members.
- The cooperation of the members may lead to the emergence of new system qualities on the macroscopic scale that cannot be reduced to their individual dynamics.
Networks of language processors

[Csuhaj–Varjú and Salomaa, 1997]
Simple eco–grammar systems

[Csuha–Varjú et al., 1997]
P2P networks

University

James  Peter  Tom

Research Institute

Mary  Jeremy
\( \Gamma = (V, (t_1, \Theta_1, \Xi_1), \ldots, (t_n, \Theta_n, \Xi_n)), n \geq 1, \) is a \( T_{ciNPMP_{F0L}} \) system, where

- \( V \) is the alphabet of the system,
- \( t_i = \{c_{i,1}, \ldots, c_{i,r_i}\}, 1 \leq i \leq n, r_i \geq 1, \) is the \( i \)-th team,
- \( c_{i,j} = (P_{i,j}, F_{i,j}, \Psi_{i,j}, \Upsilon_{i,j}), 1 \leq j \leq r_i, \) is the \( j \)-th component of the \( i \)-th team,
  - \( P_{i,j}, \) is a finite and complete set of pure context–free rules over \( V \) (i.e. rules of the form \( A \rightarrow \alpha \) with \( A \in V, \alpha \in V^* \), and for each \( A \in V, \) there is a rule \( A \rightarrow \alpha \) in \( P_{i,j} \)),
  - \( F_{i,j} \in V^\circ, \) is a non–empty finite multiset of strings (axioms),
  - \( \Psi_{i,j} = \{\psi_{i,j_1}, \ldots, \psi_{i,j_{s_i,j}}\}, \Upsilon_{i,j} = \{\upsilon_{i,j_1}, \ldots, \upsilon_{i,j_{o_{i,j}}}\}, s_i,j, o_{i,j} \geq 1, \) are finite sets of context conditions (filters) over \( V^* \),
- \( \Theta_i = \{\theta_{i1}, \ldots, \theta_{ip_i}\}, \Xi_i = \{\xi_{i1}, \ldots, \xi_{iq_i}\}, p_i, q_i \geq 1, \) are finite sets of context conditions (filters) over \( V^* \).
Configuration (state) transmissions

- Configuration (state): \( s = (M_{1,1}, \ldots, M_{1,r_1}, \ldots, M_{n,1}, \ldots, M_{n,r_n}) \),
  where \( M_{i,j} \in V^\circ, 1 \leq i \leq n, 1 \leq j \leq r_i \).
Configuration (state) transmissions

- Configuration (state): $s = (M_{1,1}, \ldots, M_{1,r_1}, \ldots, M_{n,1}, \ldots, M_{n,r_n})$, where $M_{i,j} \in V^\circ, 1 \leq i \leq n, 1 \leq j \leq r_i$.
- Rewriting step: a new string is obtained from each string in a parallel manner (through the application of Lindenmayer rules) in each multiset of strings.
Configuration (state) transmissions

- **Configuration (state):** 
  \[ s = (M_{1,1}, \ldots, M_{1,r_1}, \ldots, M_{n,1}, \ldots, M_{n,r_n}) \]
  where \( M_{i,j} \in V^\circ \), \( 1 \leq i \leq n \), \( 1 \leq j \leq r_i \).

- **Rewriting step:** a new string is obtained from each string in a parallel manner (through the application of Lindenmayer rules) in each multiset of strings.

- **Communication step:**
  - the new collections of strings sent from the current multiset of strings of the nodes either via the filters of the teams, the components or both the teams and the components satisfy the context conditions described by the given filters;
  - a new multiset of strings at a node consists of all strings that are able to penetrate the exit filter of the sender and the entrance filter of the receiver.
\( \Gamma = (V, (t_1, \Theta_1, \Xi_1), \ldots, (t_n, \Theta_n, \Xi_n)), n \geq 1, \) a \( T_{c_{rc}i_{rc}}\) NPMP\(_{FD0L}\) system:

- population growth function of \( \Gamma \): \( m(t) = \sum_{i=1}^{n} \sum_{j=1}^{r_i} \text{card}(M_{i,j}^{(t)}) \), for \( t \geq 0 \),
\( \Gamma = (V, (t_1, \Theta_1, \Xi_1), \ldots, (t_n, \Theta_n, \Xi_n)), n \geq 1, \) a \( \mathcal{T}_{\text{cr}} \mathcal{i}_{\text{rc}} \mathcal{NPMP}_{FD0L} \) system:

- population growth function of \( \Gamma \): \( m(t) = \sum_{i=1}^{n} \sum_{j=1}^{r_i} \text{card}(M_{i,j}^{(t)}) \), for \( t \geq 0 \),

- population growth function of \( \Gamma \) at node \((i, j)\): \( m_{i,j}(t) = \text{card}(M_{i,j}^{(t)}) \), for \( t \geq 0 \),
P2P communication

\[ \Gamma = (V, (t_1, \Theta_1, \Xi_1), \ldots, (t_n, \Theta_n, \Xi_n)), \ n \geq 1, \text{ a } T_{rci rc} \text{NPMP}_{FD0L} \text{ system:} \]

- population growth function of \( \Gamma \): \( m(t) = \sum_{i=1}^{n} \sum_{j=1}^{r_i} \text{card}(M_{i,j}^{(t)}) \), for \( t \geq 0 \),

- population growth function of \( \Gamma \) at node \((i, j)\): \( m_{i,j}(t) = \text{card}(M_{i,j}^{(t)}) \), for \( t \geq 0 \),

- communication function of \( \Gamma \)
  - from node \((i, j)\) to node \((k, l)\) using collective filtering:
    \[
    f_{(i,j)(k,l)}^{c}(t) = \text{card}(\{ \gamma \in M_{i,j}^{(t-1)} | \ \theta_{ix}(\gamma) = \text{true}, \xi_{ky}(\gamma) = \text{true}, 1 \leq k \leq n, 1 \leq l \leq r_k, 1 \leq x \leq p_i, 1 \leq y \leq q_k, (k, l) \neq (i, j) \}) \), for \( t = 2k', k' \geq 1 \), and \( f_{(i,j)(k,l)}^{c}(t) = 0 \) otherwise,
  - from node \((i, j)\) to node \((i, k)\) using individual filtering:
    \[
    f_{(i,j)(i,k)}^{i}(t) = \text{card}(\{ \gamma \in M_{i,j}^{(t-1)} | \ \psi_{i,ju}(\gamma) = \text{true}, \nu_{i,kv}(\gamma) = \text{true}, 1 \leq k \leq r_i, 1 \leq u \leq s_{i,j}, 1 \leq v \leq o_{i,k}, j \neq k \}) \), for \( t = 2k', k' \geq 1 \), and \( f_{(i,j)(i,k)}^{i}(t) = 0 \) otherwise.
Information dynamics

[Lázár et al., 2008], [Lázár, 2010], [Lázár, 2013]

**Theorem**

Let $\Gamma = (V, (t_1, \Theta_1, \Xi_1), \ldots, (t_n, \Theta_n, \Xi_n)), n \geq 1$, be a $T_{cr_circ}NPMP_{FD0L}$ system. Then a D0L system $H = (\Sigma, \omega, h)$ can be constructed, such that

- $m(t) = f(t)$, where $m$ is the population growth function of $\Gamma$ and $f$ is the growth function of $H$;
Let \( \Gamma = (V, (t_1, \Theta_1, \Xi_1), \ldots, (t_n, \Theta_n, \Xi_n)), n \geq 1 \), be a \( T_{cr}^{irc}\) NPMP FD0L system. Then a D0L system \( H = (\Sigma, \omega, h) \) can be constructed, such that

- \( m(t) = f(t) \), where \( m \) is the population growth function of \( \Gamma \) and \( f \) is the growth function of \( H \);
- \( m_{i,j}(t) = \text{card}(\bar{h}_{i,j}(h^t(\omega))) \) for some erasing homomorphism \( \bar{h}_{i,j} : \Sigma \to \Sigma \), where \( m_{i,j} \) is the population growth function of \( \Gamma \) at node \((i, j)\).
Theorem

(communication functions)

\[ f_{(i,j)(k,l)}^c(t) = \text{card}(\tilde{h}_{(i,j)(k,l)}(h^t(\omega))) \] for some erasing homomorphism \( \tilde{h}_{(i,j)(k,l)} : \Sigma \rightarrow \Sigma \), where \( f_{(i,j)(k,l)}^c \) is the communication function of \( \Gamma \) from node \((i, j)\) to node \((k, l)\), \( t \geq 0, 1 \leq i, k \leq n, 1 \leq j \leq r_i, 1 \leq l \leq r_k, (k, l) \neq (i, j) \), using collective filtering;
Theorem

(communication functions)

• \( f^c_{(i,j)(k,l)}(t) = \text{card}(\tilde{h}_{(i,j)(k,l)}(h^t(\omega))) \) for some erasing homomorphism \( \tilde{h}_{(i,j)(k,l)} : \Sigma \to \Sigma \), where \( f^c_{(i,j)(k,l)} \) is the communication function of \( \Gamma \) from node \((i, j)\) to node \((k, l)\), \( t \geq 0, 1 \leq i, k \leq n, 1 \leq j \leq r_i, 1 \leq l \leq r_k, (k, l) \neq (i, j) \), using collective filtering;

• \( f^i_{(i,j)(i,k)}(t) = \text{card}(\tilde{h}_{(i,j)(i,k)}(h^t(\omega))) \) for some erasing homomorphism \( \tilde{h}_{(i,j)(i,k)} : \Sigma \to \Sigma \), where \( f^i_{(i,j)(i,k)} \) is the communication function of \( \Gamma \) from node \((i, j)\) to node \((i, k)\), \( t \geq 0, 1 \leq i \leq n, 1 \leq j, k \leq r_i, j \neq k \), using individual filtering.
Relationship with D0L systems:

- Decidability problems for D0L systems:
  - The population growth function of a $T_{crei_c}NPMP_{FD0L}$ system is either exponential or polynomially bounded, which is decidable.
  - For any two $T_{crei_c}NPMP_{FD0L}$ systems, the sequence and language equivalence problems are decidable.
    - For two P2P networks it is decidable whether they function in the same manner regarding the dynamics of information.
Consequences

Relationship with D0L systems:

- **Decidability problems for D0L systems:**
  - The population growth function of a $T_{rc}^{NPMP_{FD0L}}$ system is either exponential or polynomially bounded, which is decidable.
  - For any two $T_{rc}^{NPMP_{FD0L}}$ systems, the sequence and language equivalence problems are decidable.
    - For two P2P networks it is decidable whether they function in the same manner regarding the dynamics of information.
- The alphabets of the words generated by the D0L system form an almost periodic sequence.
  - The function of these P2P networks after some time results in the saturation of information.
Discretionary Access Control (DAC)
Discretionary Access Control (DAC)

[Lázár and Farkas, 2007]

- the strings that are permitted to be sent and to be received can be defined for each peer,
Discretionary Access Control (DAC)

[Lázár and Farkas, 2007]

- the strings that are permitted to be sent and to be received can be defined for each peer,
- the filters can be used to limit traffic flow,
Discretionary Access Control (DAC)

[Lázár and Farkas, 2007]

- the strings that are permitted to be sent and to be received can be defined for each peer,
- the filters can be used to limit traffic flow,
- DAC: \((\text{subject}, \text{object}, \pm \text{access\_mode})\), where \text{subject} is the active entity permitted (denied) access to or provides an other entity with access to a resource \text{object} in the mode \text{access\_mode},
Discretionary Access Control (DAC)

[Lázár and Farkas, 2007]

- the strings that are permitted to be sent and to be received can be defined for each peer,
- the filters can be used to limit traffic flow,
- DAC: $(subject, object, \pm access\_mode)$, where $subject$ is the active entity permitted (denied) access to or provides an other entity with access to a resource $object$ in the mode $access\_mode$,
- notation: $(peer, string, \pm < direction >)$ to express DAC information flow requirements,
Discretionary Access Control (DAC)

[Lázár and Farkas, 2007]

- the strings that are permitted to be sent and to be received can be defined for each peer,
- the filters can be used to limit traffic flow,
- DAC: \((\text{subject}, \text{object}, \pm \text{access}_\text{mode})\), where \text{subject} is the active entity permitted (denied) access to or provides an other entity with access to a resource \text{object} in the mode \text{access}_\text{mode},
- notation: \((\text{peer}, \text{string}, \pm < \text{direction} >)\) to express DAC information flow requirements,
- \text{peer} corresponds to \text{subject}, \text{string} to \text{object}, and \(\pm < \text{direction} >\) defines whether the string is permitted or denied to enter (in) or leave (out) a filter of a peer,
Discretionary Access Control (DAC)

[Lázár and Farkas, 2007]

- the strings that are permitted to be sent and to be received can be defined for each peer,
- the filters can be used to limit traffic flow,
- DAC: \((\text{subject, object, } \pm \text{access_mode})\), where \text{subject} is the active entity permitted (denied) access to or provides another entity with access to a resource \text{object} in the mode \text{access_mode},
- notation: \((\text{peer, string, } \pm < \text{direction} >)\) to express DAC information flow requirements,
- \text{peer} corresponds to \text{subject}, \text{string} to \text{object}, and \pm < \text{direction} > defines whether the string is permitted or denied to enter (in) or leave (out) a filter of a peer,
- e.g., \((\text{peer, string, } +\text{in})\) and \((\text{peer, string, } +\text{out})\) means that the sender is able to transmit a string, which can be either a message or an advertisement, to the receiver.
Internet crawlers seeking novel information
Eco–foraging systems

Definition

Γ = (E, A_1, \ldots, A_n, c_{init}), \ n \geq 1, \ is \ a \ programmed \ eco–foraging \ system 
with \ appearance \ checking \ (an \ FEG_{PRac} \ system) \ such \ that 
E = (V_E, T'_E, P_E) \ is \ the \ web \ environment, 
A_i = (N_i \cup N_i^{(i)}, S_i, R_i), \ 1 \leq i \leq n, \ is \ the \ i–th \ crawler, \ a \ programmed 
grammar \ scheme \ with \ appearance \ checking, \ where 

- N_i \cup N_i^{(i)} \ is \ the \ nonterminal \ alphabet \ of \ the \ i–th \ crawler, 
- S_i \in N_i \ is \ the \ start \ symbol \ of \ the \ i–th \ crawler, \ the \ first \ web \ page \ that 
the \ crawler \ has \ to \ visit,
Eco–foraging systems (contd.)

**Definition**

- $R_i$ is a finite set of triplets of the following forms:
  - $(l_{i,1} : S_i \rightarrow S_i^{(i)}, \sigma_i(l_{i,1}), \psi_i(l_{i,1}))$, $\sigma_i(l_{i,1}) \subseteq \{l_{i,1}, \ldots, l_{i,s_i}\}$, $\psi_i(l_{i,1}) = \{l_{i,1}\}$,
  - $(l_{i,k} : X_{i,k} \rightarrow X_{i,k}^{(i)}, \sigma_i(l_{i,k}), \psi_i(l_{i,k}))$, $X_{i,k} \in N_i \setminus \{S_i\}$, $X_{i,k}^{(i)} \in N_i^{(i)} \setminus \{S_i^{(i)}\}$, $2 \leq k \leq s_i$, with $\sigma_i(l_{i,k}) \subseteq \{l_{i,1}, \ldots, l_{i,s_i}\}$, $\psi_i(l_{i,k}) \subseteq \{h_{i,2}, \ldots, h_{i,s_i}\}$, or
  - $(h_{i,k} : X_{i,k}^{(i)} \rightarrow X_{i,k}^{(i)}, \sigma_i(h_{i,k}), \psi_i(h_{i,k}))$, $X_{i,k}^{(i)} \in N_i^{(i)} \setminus \{S_i^{(i)}\}$, $2 \leq k \leq s_i$, with $\sigma_i(h_{i,k}) \subseteq \{l_{i,1}, \ldots, l_{i,s_i}\}$, $\psi_i(h_{i,k}) \subseteq \{h_{i,2}, \ldots, h_{i,s_i}\}$, where
    - $\text{Label}(R_i) = \{l_{i,1}, \ldots, l_{i,s_i}, h_{i,2}, \ldots, h_{i,s_i}\}$ is the set of labels of the rules in $R_i$.

$c_{\text{init}} = (l_{1,1}, \ldots, l_{n,1}; \omega_{\text{init}})$ is called the initial configuration of $\Gamma$, where $l_{i,1}$ is the label of the initial rule of the $i$–th crawler,

$\omega_{\text{init}} = z_1 S_{j_1} z_2 \ldots z_k S_{j_k} z_{k+1}$ is the initial state of the web environment, $S_{j_h} \in N_{j_h}$, $z_l \in V_{E}^*$, $1 \leq h \leq k$, $1 \leq l \leq k + 1$, and for some $k$, $0 \leq k \leq n$, $\{j_1, \ldots, j_k\} \subseteq \{1, \ldots, n\}$. 

Web environment

Definition

\[ E = (V_E, T'_E, \mathcal{P}_E) \] is the web environment with \( n \) foragers, \( n \geq 1 \), such that

- \( V_E \) is a finite alphabet, the alphabet of the web environment,
  \[ V_E = V_M \cup T'_E \cup V_N \cup \bar{V}_N, \] with \( V_N = \bigcup_{i=1}^{n} N_i \) and \( \bar{V}_N = \bigcup_{i=1}^{n} N_i^{(i)} \),
  where
  - \( V_M \) is a finite set,
  - \( N_i = \{X_{i,1}, \ldots, X_{i,s_i}\}, N_i^{(i)} = \{X_{i,1}^{(i)}, \ldots, X_{i,s_i}^{(i)}\}, 1 \leq s_i, 1 \leq i \leq n \), are finite alphabets,
  - \( T_E = \bigcup_{j=1}^{k} N_{i_j} \), and for some \( k, 1 \leq k \leq n \), \( \{i_1, \ldots, i_k\} \subseteq \{1, \ldots, n\} \),
  - \( T'_E = \{Z' \mid Z \in T_E\} \),
  - \( V_M, T'_E, V_N, \) and, \( \bar{V}_N \), are pairwise disjoint sets,
- \( \mathcal{P}_E = \{P_{E_1}, \ldots, P_{E_r}\} \), where \( P_{E_q}, 1 \leq q \leq r \), is a finite set of rules (the evolution rules) of the environment (insertion, deletion, etc. of webpages).
Direct derivation steps

\[ \omega_E = u_1 \alpha_i_1 u_2 \ldots u_k \alpha_i_k u_{k+1} \]

\[ \omega'_E = u_1 \beta_i_1 u_2 \ldots u_k \beta_i_k u_{k+1} \]

\[ \omega''_E = v_1 \beta_i_1 v_2 \ldots v_k \beta_i_k v_{k+1} \]
The power of eco–foraging systems

[Lázár et al., 2010], [Lázár, 2013]

**Theorem**

\[ \mathcal{L}(RE) = \mathcal{L}(\text{FEG}_{PR_{ac}}). \]

- It implies that the crawlers are able to identify any computable set of the environmental states.
The power of eco–foraging systems

[Lázár et al., 2010], [Lázár, 2013]

Theorem
\[ \mathcal{L}(\text{RE}) = \mathcal{L}(\text{FEG}_{\text{PR}_{\text{ac}}}). \]

- It implies that the crawlers are able to identify any computable set of the environmental states.
- The communication between the crawlers is minimal (indirect).
Lázár et al., 2010, Lázár, 2013

**Theorem**

$L(\text{RE}) = L(\text{FEG}_{\text{PR}_{ac}})$.

- It implies that the crawlers are able to identify any computable set of the environmental states.
- The communication between the crawlers is minimal (indirect).
- The proof of the theorem is based on the simulation of matrix grammars in the (preliminary) 2–normal form.
Eco–foraging systems with time

- Analogously to eco–foraging systems without time.
Eco–foraging systems with time

Analogously to eco–foraging systems without time.

Differences:

- During the crawling some web pages may become obsolete $\implies$ need to keep track of the aging of the web environment.
- A maximal lifetime is assigned to each web page.
- If the environment rewrites a web page and the web page will still be present in the environmental string, then the lifetime of the web page will be reduced by one regardless of whether any crawlers have managed to identify the web page or not.
- The lifetime of the newly introduced web pages is maximal.
The power of eco–foraging systems with time

[Lázár et al., 2010], [Lázár, 2013]

**Theorem**

\[ L_{\text{fin}}(\text{USC}) = L(\text{FEG}_{\text{PRac}}^{\text{time}}). \]

- It means that if the web pages may become obsolete, then the efficiency of the cooperation of the crawlers decreases considerably.
The power of eco–foraging systems with time

[Lázár et al., 2010], [Lázár, 2013]

**Theorem**

\[ \mathcal{L}_{\text{fin}}(\text{USC}) = \mathcal{L}(\text{FEG}_{\text{PRac}}^{\text{time}}). \]

- It means that if the web pages may become obsolete, then the efficiency of the cooperation of the crawlers decreases considerably.
- Proof idea: since the lifetime variable is finite, the number of nonterminals with lifetimes is also finite.
Efficiency of goal–oriented Internet crawlers in different graph topologies

Simulations to study the behaviour of the model crawlers:

- real data,
Efficiency of goal-oriented Internet crawlers in different graph topologies

Simulations to study the behaviour of the model crawlers:

- real data,
- different graph topologies:
  - scale-free worlds, scale-free small worlds, random graphs ([Erdős and Rényi, 1959], [Barabási and Albert, 1999], [Watts and Strogatz, 1998]),
Efficiency of goal–oriented Internet crawlers in different graph topologies

Simulations to study the behaviour of the model crawlers:

- real data,
- different graph topologies:
  - scale–free worlds, scale–free small worlds, random graphs ([Erdős and Rényi, 1959], [Barabási and Albert, 1999], [Watts and Strogatz, 1998]),
- comparison of the selective learning algorithm to the linear function approximation–based reinforcement learning algorithm,
Efficiency of goal–oriented Internet crawlers in different graph topologies

Simulations to study the behaviour of the model crawlers:

- real data,
- different graph topologies:
  - scale–free worlds, scale–free small worlds, random graphs ([Erdős and Rényi, 1959], [Barabási and Albert, 1999], [Watts and Strogatz, 1998]),
- comparison of the selective learning algorithm to the linear function approximation–based reinforcement learning algorithm,
- topic–specific case,
Efficiency of goal–oriented Internet crawlers in different graph topologies

Simulations to study the behaviour of the model crawlers:

- real data,
- different graph topologies:
  - scale–free worlds, scale–free small worlds, random graphs ([Erdős and Rényi, 1959], [Barabási and Albert, 1999], [Watts and Strogatz, 1998]),
- comparison of the selective learning algorithm to the linear function approximation–based reinforcement learning algorithm,
- topic–specific case,
- direct and indirect communication:
  - a positive reward is delivered to the first sender of a novel piece of information.
[Lőrincz et al., 2007]

- In the topic–specific case the relative performance of the combined learning algorithm has improved in scale–free small worlds, in scale–free worlds and in random world environment.
In the topic–specific case the relative performance of the combined learning algorithm has improved in scale–free small worlds, in scale–free worlds and in random world environment.

If the task has become more complex and the work sharing has been enforced by the environment, then the combined learning algorithm is at least equal, even superior to both the selective and the reinforcement learning algorithms in most cases.

[Lőrincz et al., 2007]
In the topic–specific case the relative performance of the combined learning algorithm has improved in scale–free small worlds, in scale–free worlds and in random world environment. If the task has become more complex and the work sharing has been enforced by the environment, then the combined learning algorithm is at least equal, even superior to both the selective and the reinforcement learning algorithms in most cases. The communication has ameliorated the performance by a large margin and adaptive communication has proven to be advantageous in the majority of the cases.
Nature–motivated (bio–inspired) computational models

Natural computing ([Ehrenfeucht et al., 2004], [Handbook of Natural Computing, 2012], [Kari and Rozenberg, 2008])

- interdisciplinary field that connects molecular biology, biochemistry, mathematics and computer science,
Nature–motivated (bio–inspired) computational models

Natural computing ([Ehrenfeucht et al., 2004], [Handbook of Natural Computing, 2012], [Kari and Rozenberg, 2008])

- interdisciplinary field that connects molecular biology, biochemistry, mathematics and computer science,
- develops and investigates models and computational techniques inspired by nature,
  - creates new and powerful computational paradigms based on the study of the structure and the functioning of natural systems,
Nature–motivated (bio–inspired) computational models

Natural computing ([Ehrenfeucht et al., 2004], [Handbook of Natural Computing, 2012], [Kari and Rozenberg, 2008])

- interdisciplinary field that connects molecular biology, biochemistry, mathematics and computer science,
- develops and investigates models and computational techniques inspired by nature,
  - creates new and powerful computational paradigms based on the study of the structure and the functioning of natural systems,
- attempts to understand the world around us in terms of information processing,
  - focuses on the interactions and the resulting properties arising in biological systems,
Background and motivation

- distributed computing systems
  - development of new technologies (P2P networks, Service–Oriented Architecture (SOA), web services, cloud computing and distributed web search),
  - increase in the amount of exchanged and published information + the rapid growth in the number of computing resources $\implies$ the need to integrate computing resources of many types into ongoing computations,
  - availability on demand, robustness and fault tolerance have to be guaranteed,
Background and motivation

- distributed computing systems
  - development of new technologies (P2P networks, Service–Oriented Architecture (SOA), web services, cloud computing and distributed web search),
  - increase in the amount of exchanged and published information + the rapid growth in the number of computing resources ⇒ the need to integrate computing resources of many types into ongoing computations,
  - availability on demand, robustness and fault tolerance have to be guaranteed,

- DNA computers may outperform silicon–based computers:
  - solution to a Hamiltonian Path Problem ([Adleman, 1994]),
  - the universal computational power of DNA rearrangements performed ciliates ([Kari et al., 1999], [Kari and Landweber, 1999]),
  - the potential for massive parallelism of DNA computers,
Background and motivation

- distributed computing systems
  - development of new technologies (P2P networks, Service–Oriented Architecture (SOA), web services, cloud computing and distributed web search),
  - increase in the amount of exchanged and published information + the rapid growth in the number of computing resources \(\Rightarrow\) the need to integrate computing resources of many types into ongoing computations,
  - availability on demand, robustness and fault tolerance have to be guaranteed,

- DNA computers may outperform silicon–based computers:
  - solution to a Hamiltonian Path Problem ([Adleman, 1994]),
  - the universal computational power of DNA rearrangements performed ciliates ([Kari et al., 1999], [Kari and Landweber, 1999]),
  - the potential for massive parallelism of DNA computers,

- establish a connection between the state–of–the–art technologies of computer science and cutting–edge research in biological science.
P2P networks, SOA, XACML

- the decentralized and ad-hoc nature of P2P networks makes the enforcement of security requirements a challenging task,
P2P networks, SOA, XACML

- the decentralized and ad–hoc nature of P2P networks makes the enforcement of security requirements a challenging task,
- multiple levels of nesting and different security needs have to be supported at varying levels of granularity (e.g. individual, unit, organization, etc.),
the decentralized and ad–hoc nature of P2P networks makes the enforcement of security requirements a challenging task,
multiple levels of nesting and different security needs have to be supported at varying levels of granularity (e.g. individual, unit, organization, etc.),
Law Governed Interaction (LGI) ([Minsky and Ungureanu, 2000], [Serban and Minsky, 2009]): model interactions between the components of large, heterogeneous systems, though, the enforcement technology for LGI has not been established yet,
the decentralized and ad–hoc nature of P2P networks makes the enforcement of security requirements a challenging task,

multiple levels of nesting and different security needs have to be supported at varying levels of granularity (e.g. individual, unit, organization, etc.),

Law Governed Interaction (LGI) ([Minsky and Ungureanu, 2000], [Serban and Minsky, 2009]): model interactions between the components of large, heterogeneous systems, though, the enforcement technology for LGI has not been established yet,

objective: to build a fully distributed access control enforcement model that is able to express eXtensible Access Control Markup Language (XACML) specifications (i.e. conflict resolution strategies, policy combination and efficient policy management) and incorporates requirements imposed by LGI–type policies in the context of P2P and SOA (entity = service).
Wireless Body Area / Body Sensor Networks (WBANs / WBSNs) – great potential in real-time monitoring of the human body (e-healthcare),
Wireless Body Area / Body Sensor Networks (WBANs / WBSNs) – great potential in real-time monitoring of the human body (e-healthcare),

security and privacy issues have to be solved,
Wireless Body Area / Body Sensor Networks (WBANs / WBSNs) – great potential in real–time monitoring of the human body (e–healthcare),

security and privacy issues have to be solved,

P2P is an emerging network technology for WBANs / WBSNs.
Internet crawlers

- limitations of the approaches to increasing the efficiency of web search in topic–specific setting:
  - not taking into account the already visited web pages ([Rungsawang and Angkawattanawit, 2005]),
  - dependence on the search engines ([Menczer, 2003]),
  - division of the web into a priori partitions before the search starts ([Risvik and Michelsen, 2002]),
Internet crawlers

- limitations of the approaches to increasing the efficiency of web search in topic-specific setting:
  - not taking into account the already visited web pages ([Rungsawang and Angkawattanawit, 2005]),
  - dependence on the search engines ([Menczer, 2003]),
  - division of the web into a priori partitions before the search starts ([Risvik and Michelsen, 2002]),

- in the Internet crawler experiments these limitations were overcome + the combined learning (weblog / selective and reinforcement learning) algorithm coupled with communication increased substantially the efficiency of web search,
Internet crawlers

- limitations of the approaches to increasing the efficiency of web search in topic-specific setting:
  - not taking into account the already visited web pages ([Rungsawang and Angkawattanawit, 2005]),
  - dependence on the search engines ([Menczer, 2003]),
  - division of the web into a priori partitions before the search starts ([Risvik and Michelsen, 2002]),

- in the Internet crawler experiments these limitations were overcome + the combined learning (weblog / selective and reinforcement learning) algorithm coupled with communication increased substantially the efficiency of web search,

- for an Internet crawler, a novel piece of information found by another crawler can serve as a context and can accelerate search,
Internet crawlers

- limitations of the approaches to increasing the efficiency of web search in topic-specific setting:
  - not taking into account the already visited web pages ([Rungsawang and Angkawattanawit, 2005]),
  - dependence on the search engines ([Menczer, 2003]),
  - division of the web into a priori partitions before the search starts ([Risvik and Michelsen, 2002]),

- in the Internet crawler experiments these limitations were overcome + the combined learning (weblog / selective and reinforcement learning) algorithm coupled with communication increased substantially the efficiency of web search,

- for an Internet crawler, a novel piece of information found by another crawler can serve as a context and can accelerate search,

- the crawlers tend to work in different compartments to avoid the overlaps between their paths and realize efficient labour division.
Microscopic reconfigurable robots

- able to reposition or even rebuild themselves to accomplish a particular task,
Microscopic reconfigurable robots

- able to reposition or even rebuild themselves to accomplish a particular task,
- typical examples: microscopic, chemical–detecting robots that move towards and metabolize pollutants,
Microscopic reconfigurable robots

- able to reposition or even rebuild themselves to accomplish a particular task,
- typical examples: microscopic, chemical–detecting robots that move towards and metabolize pollutants,
- analogy with the Internet crawlers:
  - chemicals = novel pieces of information, pollutants = obsolete ones,
  - crawlers reposition or even rebuild themselves on the basis of the learning algorithms they apply and the communication (of their weblogs).
Aims:

- to develop nature–motivated (bio–inspired) computation models, with different string manipulating operations applied in DNA computation (bio–molecular procedures performed on DNA strands, e.g. substitution, insertion, deletion, etc.) and cellular computing (gene assembly in unicellular organisms called ciliates, a process that involves reordering some fragments of DNA by permutations and possibly inversions, and deleting other fragments),
Aims:

- to develop nature–motivated (bio–inspired) computation models, with different string manipulating operations applied in DNA computation (bio–molecular procedures performed on DNA strands, e.g. substitution, insertion, deletion, etc.) and cellular computing (gene assembly in unicellular organisms called ciliates, a process that involves reordering some fragments of DNA by permutations and possibly inversions, and deleting other fragments),
- to examine what can be built by computational tools the impact of which will benefit applications that rely on state–of–the–art technologies (P2P, SOA, web services, cloud computing and distributed web search),
Aims:

- to develop nature–motivated (bio–inspired) computation models, with different string manipulating operations applied in DNA computation (bio–molecular procedures performed on DNA strands, e.g. substitution, insertion, deletion, etc.) and cellular computing (gene assembly in unicellular organisms called ciliates, a process that involves reordering some fragments of DNA by permutations and possibly inversions, and deleting other fragments),
- to examine what can be built by computational tools the impact of which will benefit applications that rely on state–of–the–art technologies (P2P, SOA, web services, cloud computing and distributed web search),
- to explore the limits of the string manipulating operations (the computational power, the robustness, the different types of complexity, the spatio–temporal dynamics and the patterns of behaviour).
to establish connections between bio–inspired paradigms and the distributed networks above:

- to choose the appropriate biological models, including the main variables of interest and interactions among them,
- to associate mathematical models with the selected biological models (discrete models, based on combinatorics, graph theory, stochastic processes to build computational models of bio–inspired systems of the distributed networks),
to establish connections between bio–inspired paradigms and the distributed networks above:
- to choose the appropriate biological models, including the main variables of interest and interactions among them,
- to associate mathematical models with the selected biological models (discrete models, based on combinatorics, graph theory, stochastic processes to build computational models of bio–inspired systems of the distributed networks),
- to investigate the complexity and the relationship between local and global structures in these discrete systems and models,
- to study the emergence of complex behaviour and the computational capabilities in systems consisting of very simple interacting components,
to establish connections between bio–inspired paradigms and the distributed networks above:

- to choose the appropriate biological models, including the main variables of interest and interactions among them,
- to associate mathematical models with the selected biological models (discrete models, based on combinatorics, graph theory, stochastic processes to build computational models of bio–inspired systems of the distributed networks),

- to investigate the complexity and the relationship between local and global structures in these discrete systems and models,
  - to study the emergence of complex behaviour and the computational capabilities in systems consisting of very simple interacting components,

- the sound computational models of natural systems allow us to employ formal reasoning about their pathways or regulatory networks, formulate predictions and / or run simulations and design novel sorts of computations based on the principles that determine the functioning of bio–inspired systems.
a biomolecular computer can be embedded into a cell, collect information from the surrounding environment (capable of learning) and regulate the cell’s behaviour (through responding to the various biological signals)
a biomolecular computer can be embedded into a cell, collect information from the surrounding environment (capable of learning) and regulate the cell’s behaviour (through responding to the various biological signals)

these molecular devices need to manipulate matter at nano-scale,
a biomolecular computer can be embedded into a cell, collect information from the surrounding environment (capable of learning) and regulate the cell’s behaviour (through responding to the various biological signals)

these molecular devices need to manipulate matter at nano–scale,

a relatively error–free, self–healing structure has to be created during the aggregation, to investigate mathematical models for self–assembly and to establish solid foundations in nanoscience:
  - to study different operations that mimic self–assembly (concatenation of strings, concatenation with overlap applied to strings, etc.),
  - to build sensors for WBANs / WBSNs and nano–fabricated (microscopic) robots via these operations,
  - to explore the limits of the operations.
A strong emphasis is placed on the following issues ([ERCIM News 85, 2011]):

- to create novel approaches, methods, algorithms and techniques for solving complex computational problems inspired by biology and provide a new basis for natural computing itself,
Summary and outlook

A strong emphasis is placed on the following issues ([ERCIM News 85, 2011]):

- to create novel approaches, methods, algorithms and techniques for solving complex computational problems inspired by biology and provide a new basis for natural computing itself,
- to utilize sophisticated computer science approaches to model various biological systems, by providing a rich set of methods and techniques to specify and analyse such systems,
A strong emphasis is placed on the following issues ([ERCIM News 85, 2011]):

- to create novel approaches, methods, algorithms and techniques for solving complex computational problems inspired by biology and provide a new basis for natural computing itself,

- to utilize sophisticated computer science approaches to model various biological systems, by providing a rich set of methods and techniques to specify and analyse such systems,

- to build possible environments where computing resources of many types are smoothly integrated into ongoing computations,
to develop architectural ideas for robust distributed systems exploiting chemical reactions and the robustness of living systems with respect to the possible failure of their components,
to develop architectural ideas for robust distributed systems exploiting chemical reactions and the robustness of living systems with respect to the possible failure of their components,

to contribute to the understanding of self-assembly, which is among the key concepts of nanoscience,
to develop architectural ideas for robust distributed systems exploiting chemical reactions and the robustness of living systems with respect to the possible failure of their components,

- to contribute to the understanding of self-assembly, which is among the key concepts of nanoscience,

- to acquire knowledge of gene regulatory networks, protein–protein interaction networks, biological transport networks and signalling pathways from the point of view of information processing, which assists in harnessing the cell as a programmable "nano–bot", to carry on specific tasks such as targeted drug delivery, housekeeping of chemical factories and coordination of biofilm scaffolding and self–assembling.
References I


References II


References III


References IV


Thank you for your attention!