

**Universitatea Tehnică a Moldovei**

**FCIM  
DIIS**

# **Evaluarea performantelor**

Titular:

Conf.univ.,dr. V. Ababii

# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

## Inteligenta Artificiala:

- Retele Neuronale;
- Logica Fuzzy;
- Retele Petri. In combinatie cu alte modele din IA;
- Algoritmi genetici. Calcul Evolutiv;
- Calcul Membrantar sau P-Systems;
- Inteligenta de roi (albine, furnici, stol, turma, colonie, pride etc.);
- Calcul celular;
- Calcul inspirat din natura.

# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

**Retele Neuronale**

**Logica Fuzzy**

**Retele Petri**

# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Algoritmi Genetici. Calcul Evolutiv.

**Calcul evolutiv** = dezvoltarea și utilizarea de tehnici de rezolvare a problemelor inspirate de evoluția speciilor în natură.

**Sursa de inspirație: teoria evoluției speciilor biologice** =

- Populațiile evoluează prin apariția de noi caracteristici ale indivizilor în timpul încrucișării și ca efect al mutațiilor aleatoare;
- În procesul de evoluție supraviețuiesc indivizii care se adaptează cel mai bine mediului;

**Rezolvarea unei probleme** =

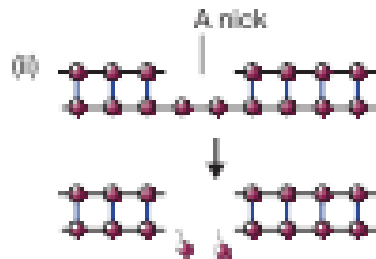
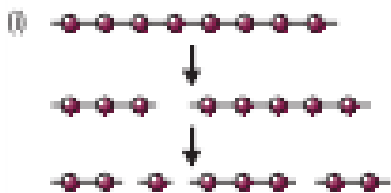
- căutarea soluției în spațiul tuturor soluțiilor potențiale folosind o populație de agenți (căutători);
- căutarea este ghidată prin intermediul unei funcții care măsoară gradul de apropiere față de soluție.

# Algoritmi Genetici si P-systems

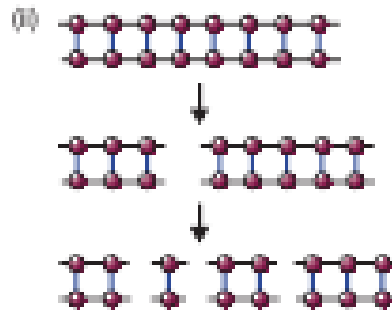
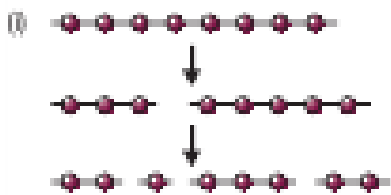
Metode si modele bazate pe Inteligenta Artificiala

## Algoritmi Genetici. Calcul Evolutiv.

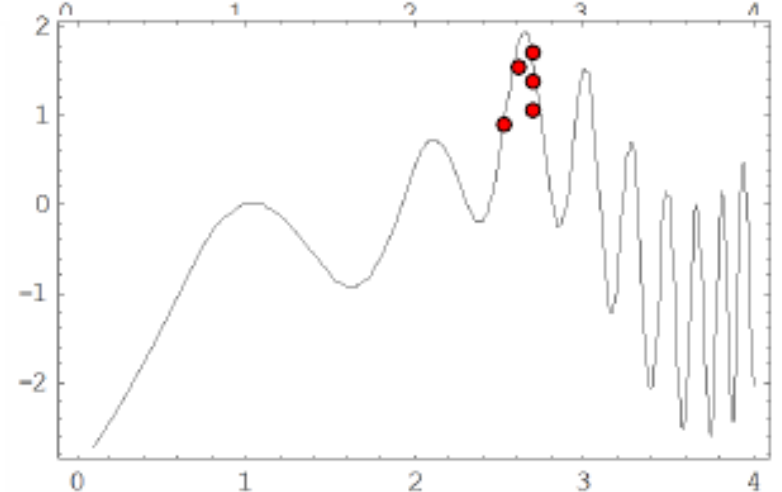
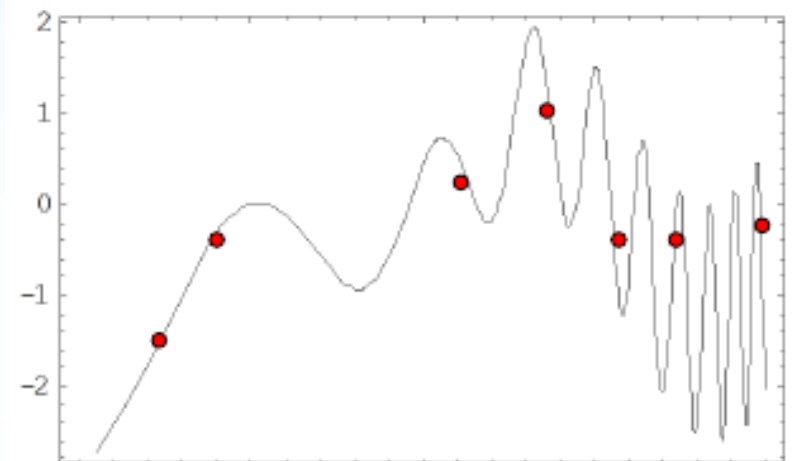
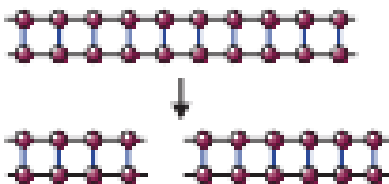
(a) S1 nuclease



(b) DNase I



(c) A restriction endonuclease



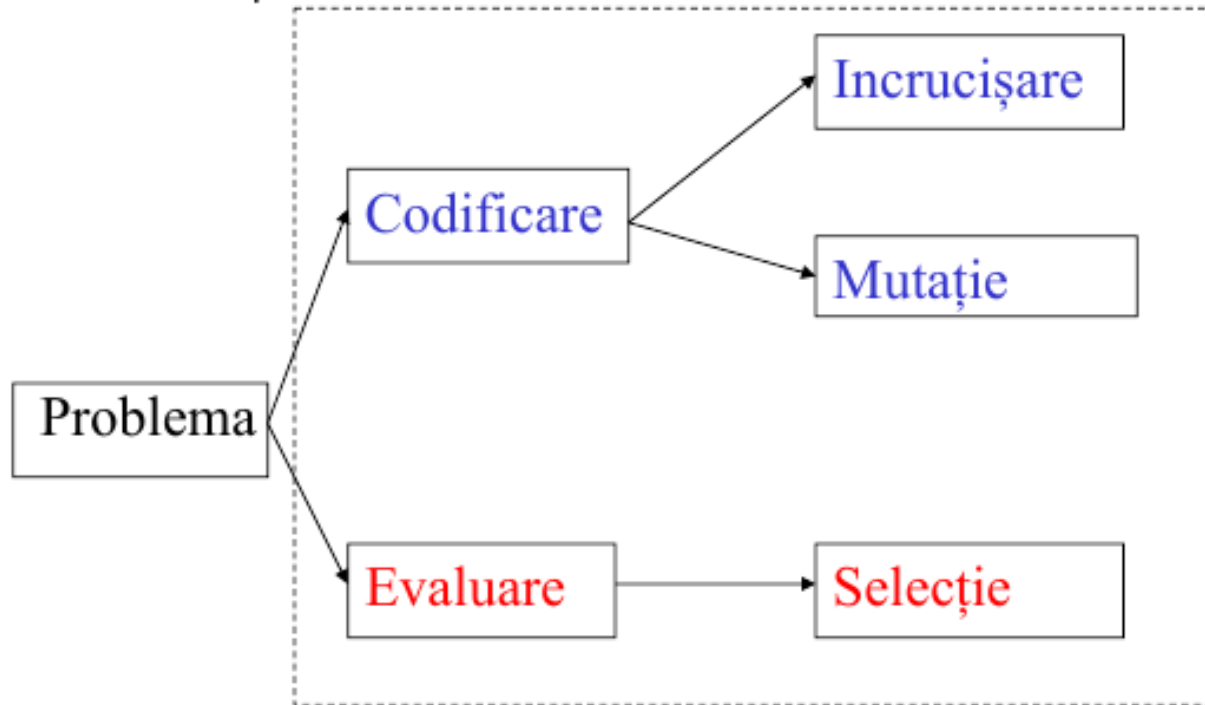
# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

**Algoritmi Genetici. Calcul Evolutiv.**

## Proiectarea unui algoritm evolutiv

Elemente componente:



# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

**Algoritmi Genetici. Calcul Evolutiv.**

## Structura unui algoritm evolutiv

Inițializare populație (aleatoare)

Evaluare populație (calcul fitness pt fiecare element)

REPEAT

**Selecție** părinți

    Generarea urmașilor prin **încrucișare**

    Modificarea urmașilor prin **mutație**

**Evaluare** urmași

**Selecție** supraviețuitori (vor forma populația din noua generație)

UNTIL <condiție de oprire>

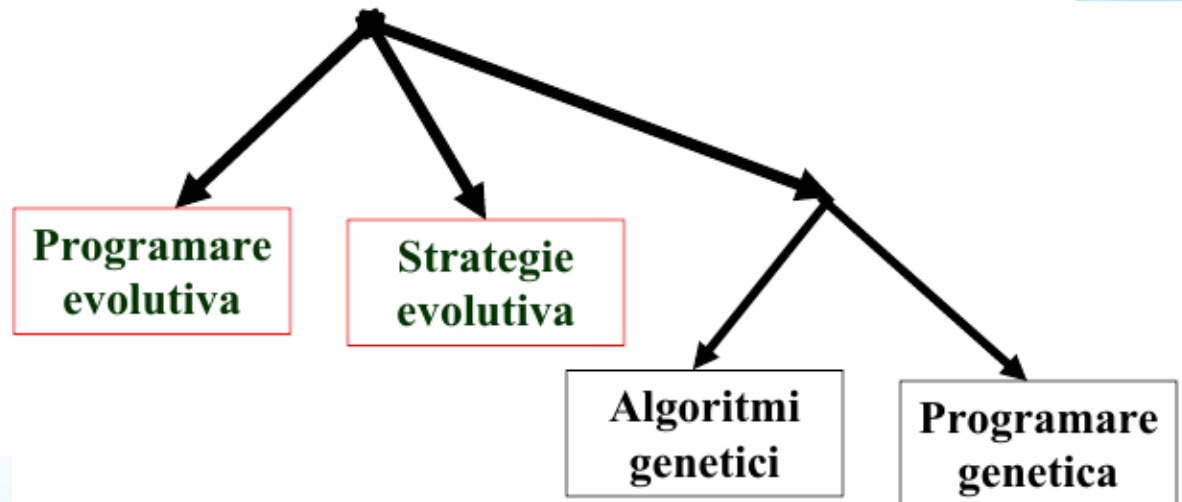
# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

**Algoritmi Genetici. Calcul Evolutiv.**

## Algoritmi genetici

- Codificarea elementelor populației
- Construirea funcției fitness (funcție scor, funcție de adecvare)
- Selecție
- Incruciașare
- Mutație





# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Algoritmi Genetici. Calcul Evolutiv.

**Algoritmii genetici** sunt tehnici adaptive de căutare euristică, bazate pe principiile geneticii și ale selecției naturale, enunțate de Darwin ("supraviețuiește cel care e cel mai bine adaptat"). Mecanismul este similar procesului biologic al evoluției. Acest proces posedă o trăsătură prin care numai speciile care se adaptează mai bine la mediu sunt capabile să supraviețuiască și să evolueze peste generații, în timp ce acelea mai puțin adaptate nu reușesc să supraviețuiască și cu timpul dispar, ca urmare a selecției naturale. Probabilitatea ca specia să supraviețuiască și să evolueze peste generații devine cu atât mai mare cu cât gradul de adaptare crește, ceea ce în termeni de optimizare înseamnă că soluția se apropie de optim.

Un algoritm genetic reprezintă o transpunere în programare a unor principii biologice. Structura unui astfel de algoritm este descrisă prin următorii pași:

*Pasul 1.* generează populația inițială

*Pasul 2.* generează urmașii (*off-spring*)

*Pasul 3.* aplică mutații genetice

*Pasul 4.* selectează o parte a populației care părăsește sistemul

*Pasul 5.* repetă pașii 2, 3, 4, până la satisfacerea unui criteriu de optimalitate (sau de un anumit număr de ori)

# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Algoritmi Genetici. Calcul Evolutiv.

### Populația

Accasta este, din punct de vedere matematic, mulțimea soluțiilor posibile ale problemei. De exemplu, pentru cazul în care se modelează o problemă de minim (sau de maxim) pentru o funcție  $f: \mathbb{R}^n \rightarrow \mathbb{R}$ , atunci orice obiect  $(x_1, x_2, \dots, x_n) \in \mathbb{R}^n$  reprezintă o soluție posibilă. În consecință, în acest caz, orice submulțime  $P \subseteq \mathbb{R}^n$  reprezintă, din punctul nostru de vedere, *populația* unui algoritm genetic.

### Indivizii

Populația este formată din indivizi; din punct de vedere matematic, indivizii sunt elementele mulțimii  $\Pi$ . La rândul său, fiecare individ este caracterizat de trăsăturile sale, de zestrea sa cromozomială. Astfel, revenind la exemplul problemei de optimizare, este ușor de observat că un individ (adică un punct din  $\mathbb{R}^n$ ) este complet determinat de coordonatele sale, respectiv de valorile  $x_1, x_2, \dots, x_n$ . Desigur, fiecare coordonată va influența valoarea funcției care se minimizează (maximizează).

# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Algoritmi Genetici. Calcul Evolutiv.

**Urmașii** (generația nouă, *off-spring*)

Această noțiune modelează un procedeu de modificare (îmbogățire) a mulțimii  $\Pi$ . Putem înțelege mai bine această situație dacă o comparăm cu alte metode de calcul. De exemplu, anumite metode numerice determină soluția prin intermediul unui șir de valori (definite, eventual, iterativ) ceea ce reprezintă, de asemenea, o situație de îmbogățire a unei mulțimi, prin construcția, succesivă, de noi valori. De asemenea, metodele de simulare cunosc modalități de creare a de noi entități, adecvate unei anumite probleme.

Modul de generare a urmașilor depinde, intrinsec, de problema modelată, fără să existe un procedeu general. Astfel, pentru problema de optimizare a funcției  $f$  descrise mai sus, putem considera ca un urmaș provine din alăturarea componentelor (trăsăturilor cromozomiale) preluate de la doi “părinți”. Matematic, dacă vom considera doi indivizi:

$$(x_1, x_2, \dots, x_k, x_{k+1}, \dots, x_n)$$

$$(y_1, y_2, \dots, y_k, y_{k+1}, \dots, y_n)$$

atunci, printr-un procedeu inspirat din mecanismul *cross-over*, putem imagina doi urmași, de forma:

$$(x_1, x_2, \dots, x_k, y_{k+1}, \dots, y_n)$$

$$(y_1, y_2, \dots, y_k, x_{k+1}, \dots, x_n)$$

obținuți prin “divizarea lanțului cromozomial” în poziția  $k$ .

# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

## Algoritmi Genetici. Calcul Evolutiv.

### Mutații genetice

Orice modificare “*suficient de mică*” a unui “*cromozom*” reprezintă o mutație genetică. Astfel de mutații pot fi haotice (întâmplătoare) sau pot fi urmarea unor așa-numite achiziții genetice. Desigur, modelul cel mai adecvat de mutație genetică va putea fi ales numai în funcție de problema modelată. În cazul particular al unei probleme de optimizare, cea mai potrivită cale de reprezentare a mutațiilor genetice este constituită de așa numita deplasare pe direcția gradientului. Justificarea unui astfel de procedeu se regăsește în metodele numerice cunoscute, de tip *Newton*, în care determinarea soluției se face printr-o deplasare pe o *direcție* care ține cont de gradientul funcției.

# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Algoritmi Genetici. Calcul Evolutiv.

### Selecția

Acest pas reprezintă modul în care anumiți indivizi ies din sistem. Alegerea unui procedeu de ieșire din sistem ține de problema modelată. Astfel, este cu totul natural ca pentru o problemă de optimizare, clasificarea populației să se facă în funcție de valoarea corespunzătoare a funcției obiectiv. Această funcție devine, necesar, o funcție de performanță, prin intermediul căreia se va decide ieșirea din sistem a indivizilor.

Concret, pentru o problemă de minimizare a unei funcții  $f$ , vom clasifica indivizii *crescător*, în funcție de valoarea corespunzătoare a funcției  $f$  și vom elimina surplusul de populație, rezultat ca urmare a generării de „urmasi”. Pentru o problemă de maximizare vom clasifica indivizii în ordinea descrescătoare a valorilor funcției obiectiv.

# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Algoritmi Genetici. Calcul Evolutiv.

### Finalitatea

Evident, ca în orice algoritm, trebuie să existe un test de terminare a acestuia. Finalitatea este determinată de scopul urmărit, deci nu se poate indica o strategie generală. Pentru probleme de optimizare, așa cum am arătat în lucrarea [2] se poate folosi un test similar cu estimarea *a posteriori* a erorii, cunoscută în probleme de calcul numeric. Astfel, dacă notăm prin  $x_n$  individul cu cea mai bună performanță al generației a  $n$ -a, atunci un test de oprire a algoritmului poate fi de forma:

$$|f(x_{n-1}) - f(x_n)| < \varepsilon$$

# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Algoritmi Genetici. Calcul Evolutiv.

Considerăm următoarea problemă de optimizare:

$$\min f(x, z, y)$$

$$f(x, z, y) = (x-1)^2 + (y-2)^2 + (z-4)^2$$

**Populația.**

**Funcția de performanță.**

**Inițializarea.**

# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Algoritmi Genetici. Calcul Evolutiv.

**Mutațiile genetice.** După cum am menționat, mutațiile genetice joacă un rol cheie în algoritm, deoarece, așa cum am văzut în [2] de îndată ce inițiem mutații genetice pe direcția gradientului funcției obiectiv, algoritmul este convergent. La rândul său, gradientul poate fi calculat folosind o formulă de calcul aproximativ, pentru fiecare derivată parțială:

$$\frac{\partial f}{\partial x} \approx \frac{f(x+h, y, z) - f(x, y, z)}{h}$$

$$\frac{\partial f}{\partial y} \approx \frac{f(x, y+h, z) - f(x, y, z)}{h}$$

$$\frac{\partial f}{\partial z} \approx \frac{f(x, y, z+h) - f(x, y, z)}{h}$$



# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Algoritmi Genetici. Calcul Evolutiv.

Tehnica de programare genetică poate fi aplicată pentru determinarea soluțiilor sistemelor de ecuații neliniare. Pentru un astfel de sistem, scris formal:

$$\begin{cases} F_1(x_1, x_2, \dots, x_n) = 0 \\ F_2(x_1, x_2, \dots, x_n) = 0 \\ \dots \\ F_m(x_1, x_2, \dots, x_n) = 0 \end{cases}$$

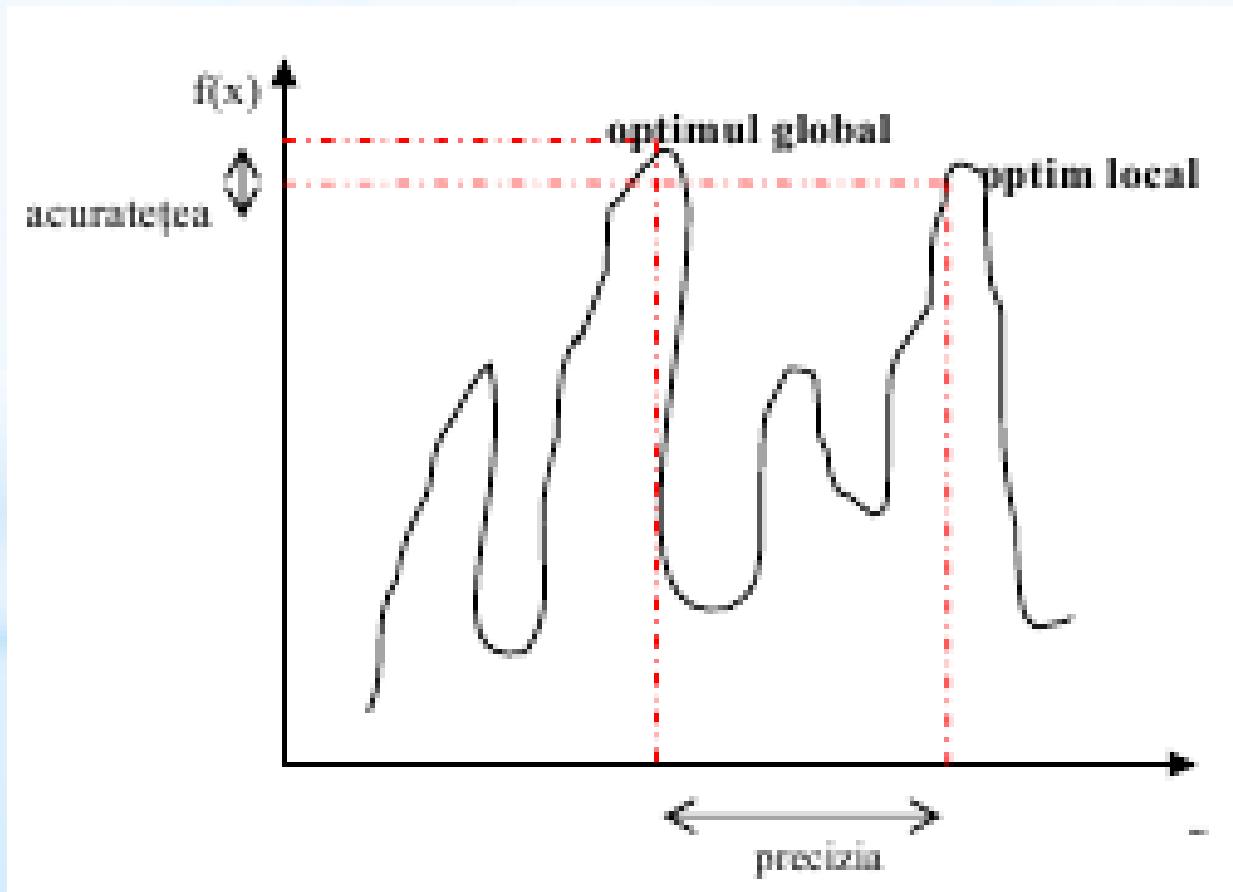
vom considera  $f(x_1, x_2, \dots, x_n) = \sum_{i=1..m} (F_i(x_1, x_2, \dots, x_n))^2$ , iar sistemul de ecuații dat este echivalent cu problema de optimizare:

$$\min f(x_1, x_2, \dots, x_n)$$

# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

## Algoritmi Genetici. Calcul Evolutiv.

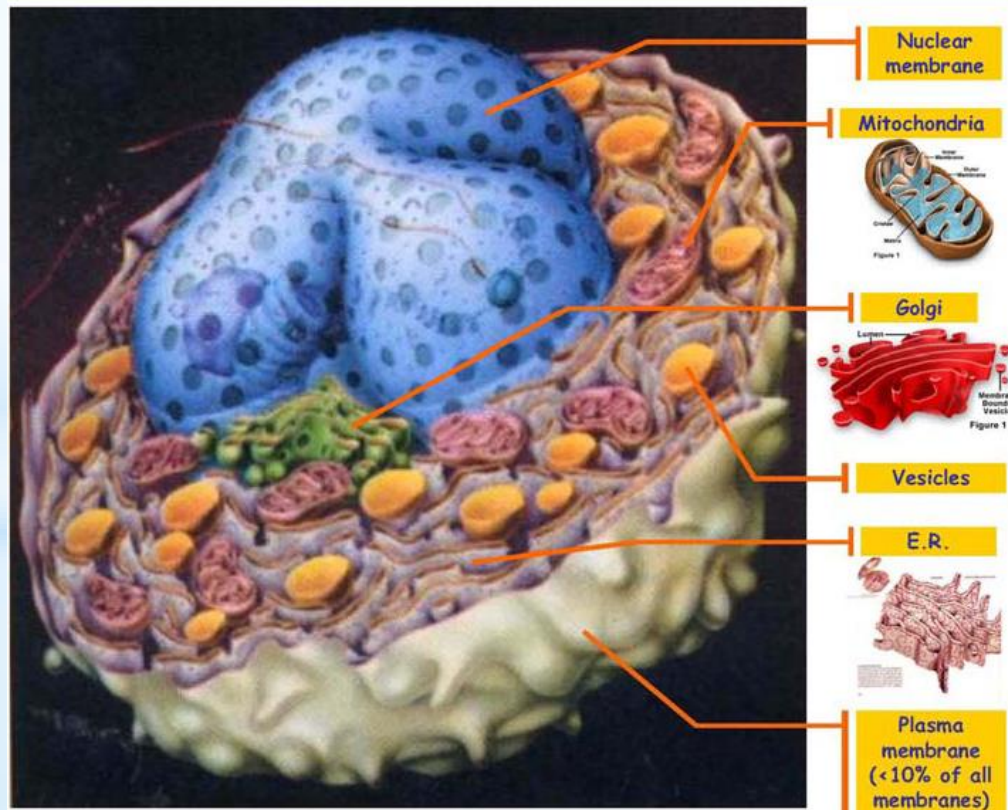


# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

## Calcul Membrantar. P-Systems.

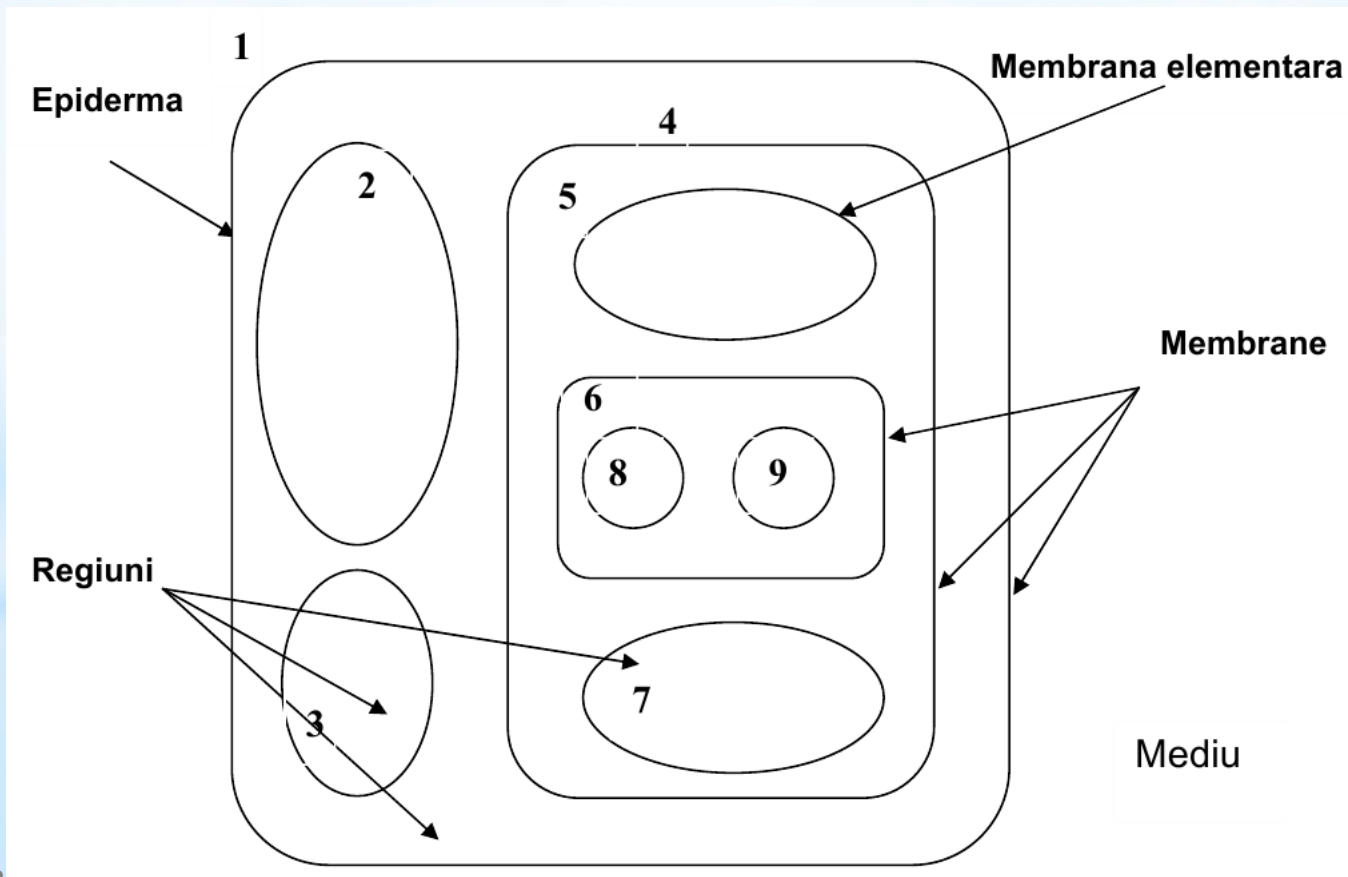
### Structural Architecture



# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

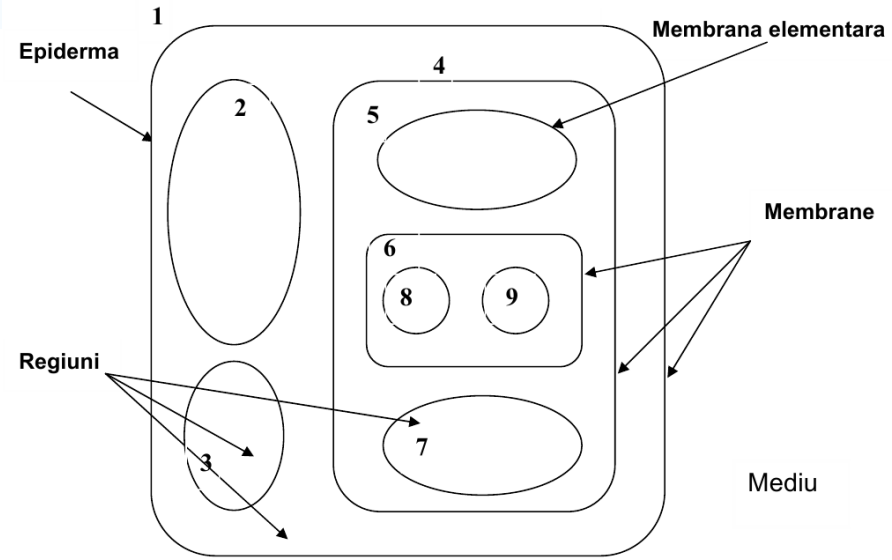
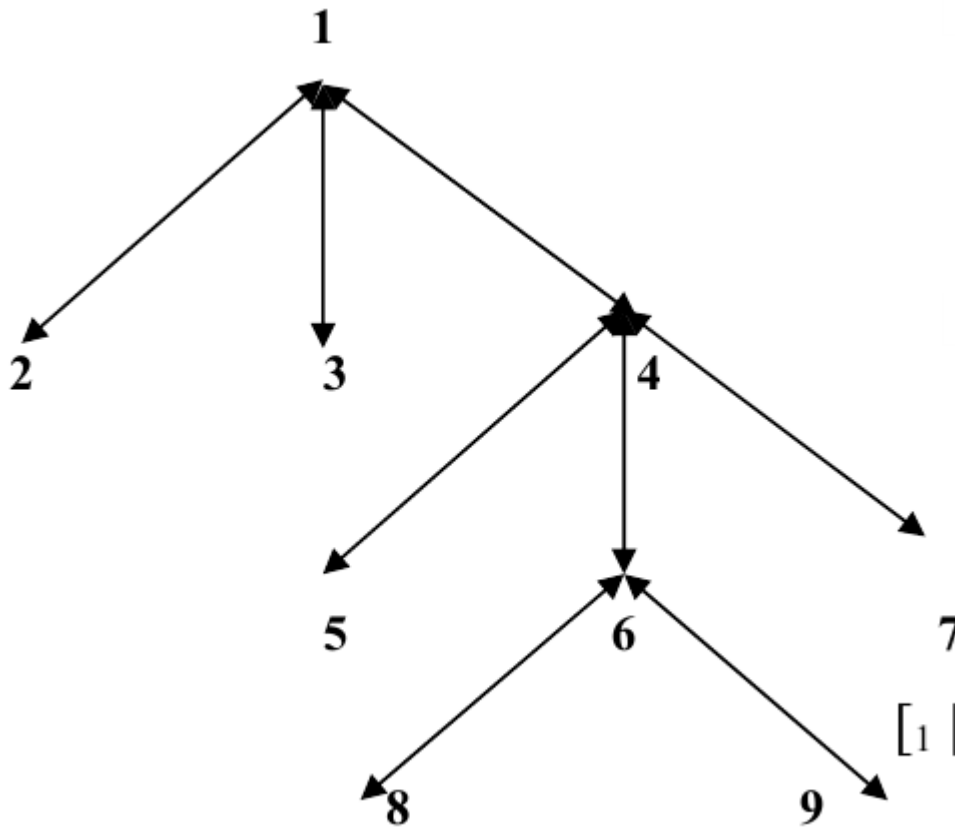
## Calcul Membrantar. P-Systems.



# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

## Calcul Membrantar. P-Systems.



[1 [2 ]2 [3 ]3 [4 [5 ]5 [6 [8 ]8 [9 ]9 ]6 [7 ]7 ]4]1

# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

## Calcul Membranar. P-Systems.

$$\pi = (O, C, \mu, w_1, w_2, w_3, \dots, w_m, R_1, R_2, \dots, R_m, i_0)$$

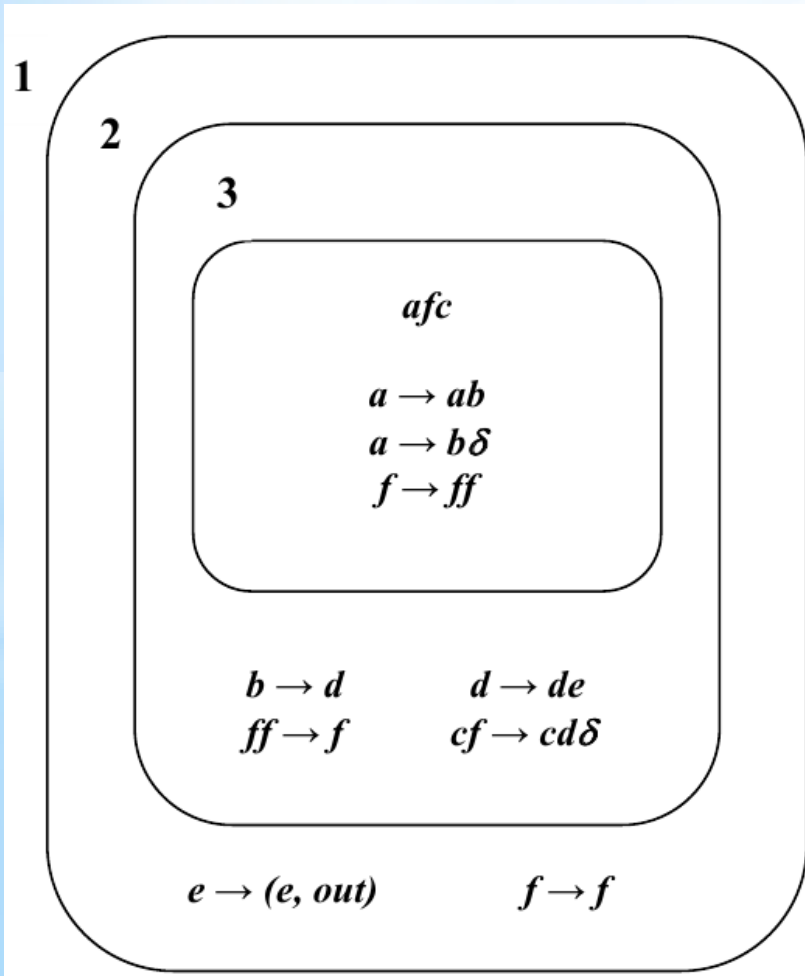
unde:

1.  $O$  este alfabetul finit si nevid de obiecte
2.  $C \subseteq O$  este multimea catalizatorilor
3.  $\mu$  este structura membranara, fomata din  $m$  membrane, notate cu  $1, 2, \dots, m$ ; de unde si denumirea de structura membranara si sistem de grad  $m$ ;
4.  $w_1, w_2, w_3, \dots, w_m$  sunt cuvinte peste alfabetul  $O$  reprezentand multimea obiectelor prezente in regiunile  $1, 2, \dots, m$  ale structurii membranare.
5.  $R_1, R_2, \dots, R_m$  sunt multimi finite de reguli ale evolutiei asociate regiunilor  $1, 2, \dots, m$  ale structurii membranare.
6.  $i_0$  este fie unul dintre indicii  $1, 2, \dots, m$  si atunci regiunea respectiva este zona de output a sistemului, sau fie este  $\emptyset$ , in care caz rezultatul fiind colectat in mediul inconjurator sistemului.

# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

## Calcul Membrantar. P-Systems.



$$\left\{ \begin{array}{l} R_1 : ca^3 \rightarrow cx, \\ R_2 : ca \rightarrow z, \end{array} \right.$$

$$\left\{ \begin{array}{l} R_1 : a^3 \rightarrow x, \\ R_2 : a \rightarrow z, \\ R_1 > R_2, \end{array} \right.$$

$$\left\{ \begin{array}{l} R_1 : a^3 \rightarrow x, \\ R_2 : a \rightarrow z, \\ R_3 : [1d]_1 \rightarrow [1p]_1[1q]_1, \\ R_4 : zp \rightarrow p, \\ R_5 : p \rightarrow \lambda, \\ R_6 : xq \rightarrow q, \\ R_7 : q \rightarrow \lambda, \\ R_1 > R_2 > R_3; R_4 > R_5; R_6 > R_7. \end{array} \right.$$

# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Swarm Intelligence.

**Inteligența „roi”** (*Swarm intelligence*), numită și inteligență colectivă sau de grup, reprezintă o abordare a inteligenței artificiale asemenea rețelelor neuronale. În acest caz programatorii studiază și modelează comportarea inteligentă la sisteme naturale ca de exemplu inteligența microbiană, creșterea bacteriilor, la roiurile de albine sau coloniile de furnici, chiar dacă, la nivel individual, fiecare în parte nu e neapărat "inteligent", ci doar urmează reguli simple. Se urmăresc și se analizează relații din mediul natural, cum ar fi relația pradă-prădător. În acest fel se poate deduce modul cum funcționează inteligența într-un grup de vietăți sau și sisteme, plecând de la reguli simple și ajungând la un nivel de comportament individual. Se ajunge la realizarea și dezvoltarea unor sisteme inteligente creând programe agent care imită comportamentul acestor sisteme naturale.



# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Calcul inspirat din natură.

Swarm Intelligence – comportamentul inteligent al mulțimilor (inteligența roiurilor)

- **ACO - Ant Colony Optimization** - Modelul coloniei de furnici
- **PSO - Particle Swarm Optimization** - Modelul ansamblului de particule (sau a stolului de păsări)
- **ABC - Artificial Bee Optimization** – Modelul roiului de albine

# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Calcul inspirat din natură.

**Swarm intelligence** = domeniu care cuprinde tehnici inteligente bazate pe comportamentul colectiv al unor sisteme cu auto-organizare și fără control centralizat

Termen introdus în 1989 de Gerardo Beni și Jing Wang în contextul sistemelor de roboți

Tehnicile din “swarm intelligence” se bazează pe mulțimi de agenți caracterizați prin:

- Reguli simple de “funcționare”
- Interacțiuni locale
- Absența unor structuri de control centralizat

# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Calcul inspirat din natură.

Exemple de sisteme naturale având astfel de caracteristici:

- Colonii de furnici
- Roiuri de albine
- Stoluri de păsări
- Bancuri de pești



Reprezintă modele pentru tehnici de rezolvare a unor probleme de optimizare sau de analiză a datelor



# Algoritmi Genetici și P-systems

Metode și modele bazate pe Inteligența Artificială

## Calcul inspirat din natură. Modelul coloniei de furnici

Problema comis voiajorului

Reprezentarea soluției:  $(i_1, i_2, \dots, i_n)$  permutare a mulțimii de indici ai orașelor

**Probabilități de tranziție**

(furnica  $k$  trece la momentul  $t$  de la orașul  $i$  la orașul  $j$ )

$$P^k(i, j, t) = \begin{cases} \frac{(\tau_{ij}(t))^\alpha (\omega_{ij}(t))^\beta}{\sum_{l \in V(i, k)} (\tau_{il}(t))^\alpha (\omega_{il}(t))^\beta} & V(i, k) \neq \emptyset \\ 0 & \text{altfel} \end{cases}$$

Concentrația de feromon corespunzătoare arcului  $(i, l)$

Factor invers proporțional cu costul arcului  $(i, l)$

Lista orașelor nevizitate încă de furnica  $k$  și care sunt conectate cu orașul  $i$

# Algoritmi Genetici si P-systems

Metode si modele bazate pe Inteligenta Artificiala

**Calcul inspirat din natura. Modelul coloniei de furnici**

## Colonia de furnici

- ❑ Furnica
- ❑ Cuib (furnici de acelasi tip)
- ❑ Tip de miros
- ❑ Intalnirea a doua furnici
- ❑ Crearea unui cuib
- ❑ Migrarea furnicilor intre cuiburi
- ❑ Eliminarea unei furnici din cuib

## Proces de grupare a datelor

- ❑ Data
- ❑ Cluster (clasa de date similare)
- ❑ Prag de similaritate
- ❑ Compararea a doua date
- ❑ Initierea unui cluster
- ❑ Transfer de date de la un cluster la altul
- ❑ Eliminarea unei date dintr-un cluster

# Algoritmi Genetici si P-systems

**Inteligența artificială va sta la dezvoltarea sistemelor pentru viitor**