BABEŞ-BOLYAI UNIVERSITY
Faculty of Mathematics and Computer Science

Inteligenţă Artificială

10: Modele Hibride si Aplicatii

Camelia Chira

camelia.chira@ubbcluj.ro

Am vazut deja ca...

- **Algoritmi evolutivi**
  - **Reprezentare**
  - **Functia de evaluare** sau de **fitness**
  - Operatori de **variatie: incrucisare, mutatie**
  - **Selectia**
- **Algoritmi inspirati de natura (swarm intelligence)**
  - **Particle Swarm Optimization (PSO)**
  - **Ant Colony Optimization (ACO)**
- **Algoritmi de invatare**
  - **Arbori de decizie**
  - **Retele neuronale**

Astazi:

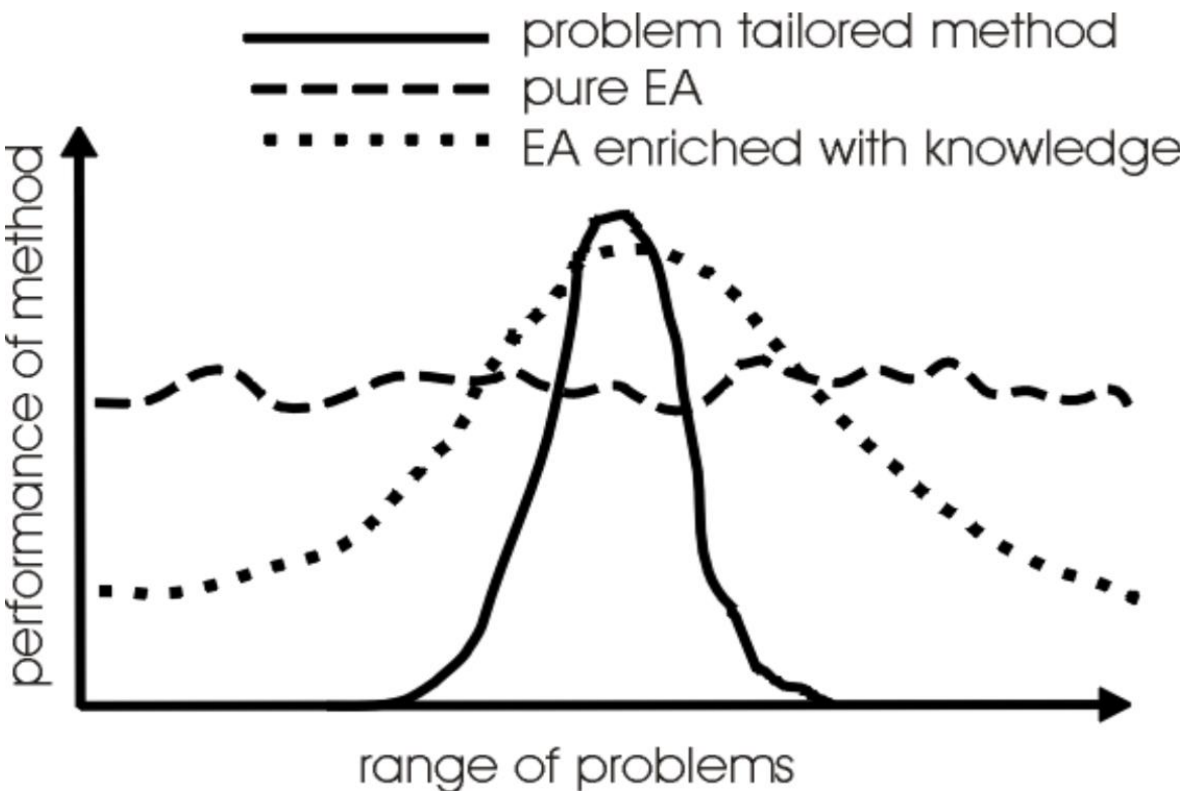
- Modele hibride

Modele hibride

- **Motivatie:** Un algoritm (ex. AE) poate face parte dintr-un model / sistem mai mare

- Imbunatatirea
  - Unei tehnici existente cu elemente AE
  - Unui AE pentru o mai buna cautare a solutiilor interesante

EA Performance



MICHALEWICZ

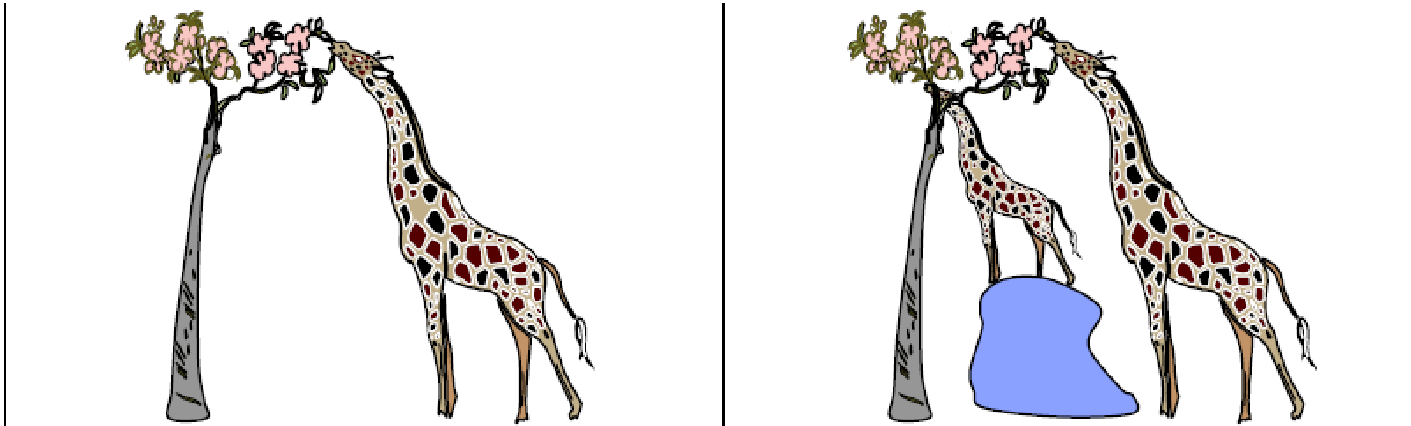
Modele hibride

- Hibridizarea AE cu proceduri standard (Ex. Hill-climbing, metode de cautare greedy)
  - Populatia initiala poate fi generata folosind alte metode
  - Indivizii pot trece printr-o etapa de imbunatatire locala si apoi intra in competitie cu ceilalti indivizi din populatie
  - Etc ....hybrid systems with fuzzy-neural-evolutionary components!
- Exemple
  - 2-opt local search in EA pentru TSP
  - EA + Hill-climbing
  - EA + Tabu search
  - Simulated Annealing + EA
  - ES + Local search
  - Greedy crossover, local search for mutation
  - Local search + Simulated Annealing
  - etc

Algoritmi memetici (Memetic Algorithms)

Algoritmi evolutivi + Operatori specifici de cautare locala

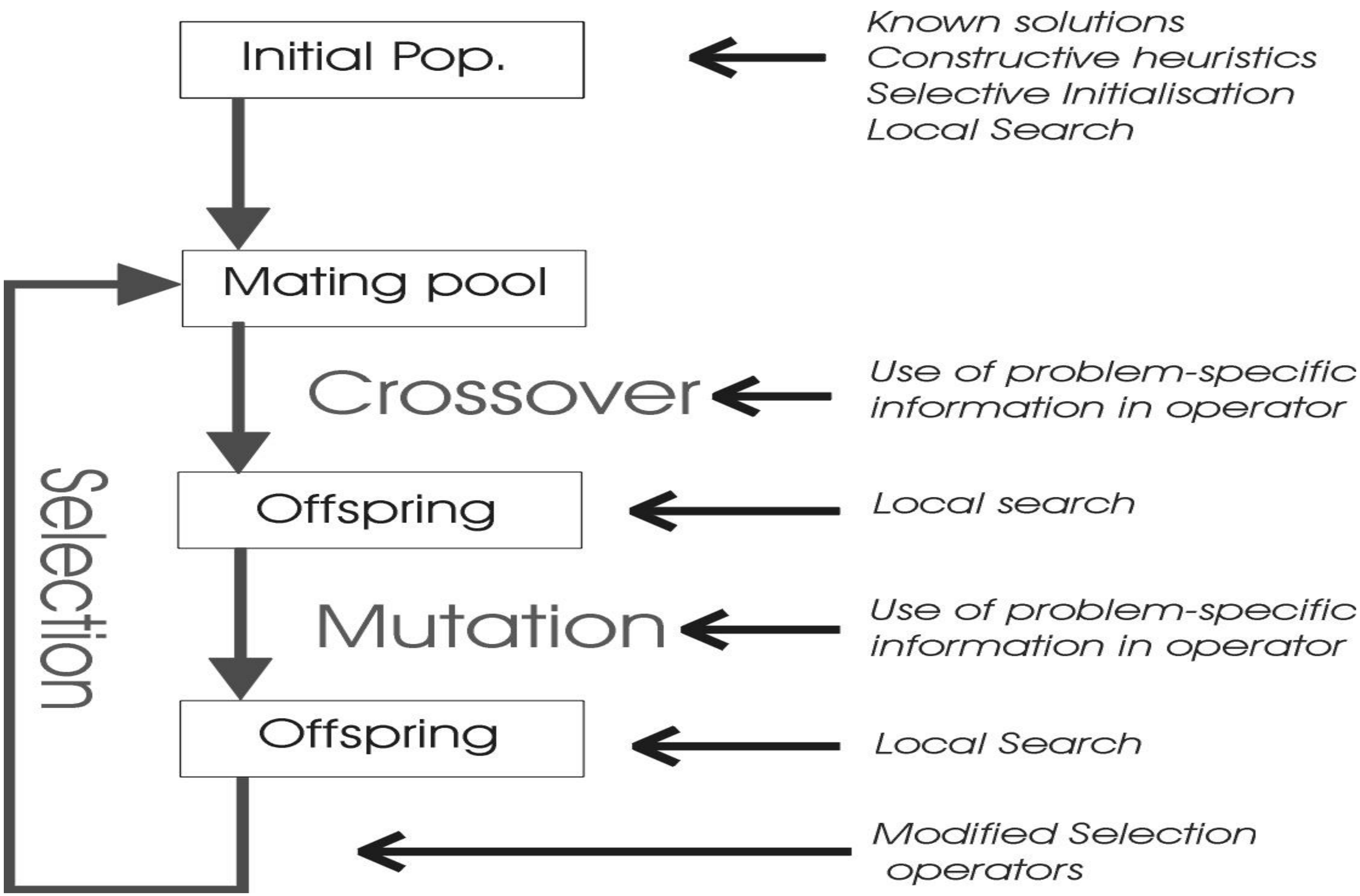
- Mai eficienti si capabili sa ofere solutii de acuratete mai buna in comparatie cu EA pentru anumite probleme
- local search is applied during the evolutionary cycle



a) Genetic algorithms: Survival of the genetically fittest (i.e., tallest)

b) Memetic algorithms: Survival of the genetically fittest and most experienced

Hibridizare – unde?



Eiben & Smith

Euristici pentru initializarea populatiei

- *n*-way tournament intre solutii generate aleator => populatia initiala

- Multi-Start Local Search
  - Alege aleator n puncte din care sa porneasca un hill-climbing

- Initializarea populatiei cu solutii deja cunoscute sau gasite folosind o alta tehnica
  - De obicei creste performanta medie, populatia fiind biased spre solutii cunoscute
  - Cea mai buna performanta vine insa tot de la solutii aleatoare (cf. unor experimente)

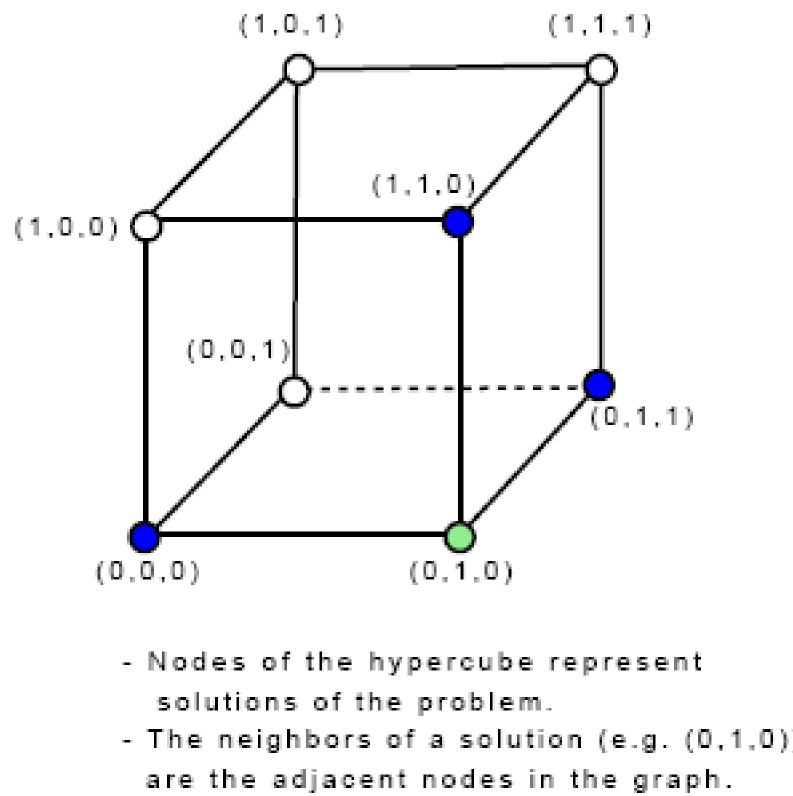
Operatori specifici

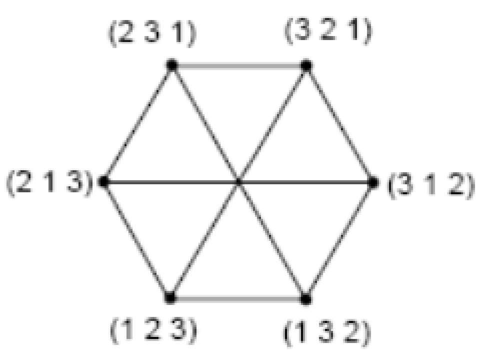
- Folosirea unor informatii specifice problemei in definirea unor operatori ‘inteligenti’

- Exemple
  - Operator specific pentru TSP: mostenirea unor subtururi comune din parinti si conectarea lor folosind o euristica nearest neighbour (DPX)
  - Operatorul ‘Pull Move’ pentru modelul HP de proteine in problema de prezicere a structurii proteinelor

Cautare locala pentru un offspring

- Poate fi privita ca si invatare de-a lungul vietii
- Stabilirea unei notiuni de vecinatate pentru fiecare solutie: x <- N(x)
  - N(x) contine toate punctele in care se poate ajunge din x printr-o singura aplicare a unui operator (move operator)





Vecinatate pentru permutari. Ex. Vecinii (2 3 1) sunt (2 1 3), (3 2 1), (1 3 2)

Regulile cautarii locale (Pivot rules)

- Cum se efectueaza cautarea in N(x): aleator, sistematic sau exhaustiv

- Greedy Ascent
  - Cautarea se opreste imediat ce un vecin cu un fitness mai bun a fost gasit
- Steepest Ascent
  - Toti vecinii sunt evaluati si cel mai bun este ales

- Variatii ale cautarii locale:
  - Cautarea locala se aplica intregii populatii?
    - Numai celei mai bune solutii?
    - Numai celei mai slabe solutii?
  - Cate iteratii de cautare locala pentru un individ?

TSP: Integrarea unor metode de cautare locala

- Efort mare in a gasi reprezentari TSP + operatori de variatie
- O solutie posibila: *Hibridizari intre EAs si alte metode de cautare*

begin
    $t \leftarrow 0$
    Initializare P(t)
    Aplicare cautare locala pentru P(t)
    Evaluare P(t)
    while (not termination-condition) **do**
    begin
        $t \leftarrow t + 1$
        Select P(t) din P(t-1)
        Modificare P(t)
        Aplicare cautare locala pentru P(t)
        Evaluare P(t)
    end
end

2-opt
3-opt
Lin-Kernighan
etc
(alti operatori single- sau multi-parent)

Integrarea unor metode de cautare locala

- Putem folosi solutia imbunatatita local in recombinari

Exemplu hibridizare AE-LS:

1. Folosim 2-opt pentru a inlocui fiecare ruta din populatia curenta cu o ruta optima local
2. Indivizii care au un fitness mai bun sunt selectati mai des pentru recombinare
3. Avem incrucisare si mutatie
4. Cautarea locala este aplicata fiecarui individ
5. Repeta pasii 2,3,4 pana cand o conditie de terminare e satisfacuta.

Modele de adaptare

- **Lamarckian**
  - Caracteristici dobandite de un individ in timpul vietii pot fi transmise la descendenti

    Ex. Inlocuieste un individ cu vecinul mai bun

- **Baldwinian**
  - Caracteristici dobandite de un individ nu pot fi transmise la descendenti

    Ex. Individul primeste fitness-ul dar nu si genotipul celui mai bun vecin

The Baldwin effect
MA – nu suntem constransi de realitati biologice =>
Putem aplica Lamarckian learning

Diversitatea populatiei

- Mentinerea diversitatii poate fi o problema in EAs, mai ales cand cautarea locala tinde spre cateva solutii foarte bune

- Moduri de a induce diversitatea
  - Alege indivizii care trec prin faza de cautare locala in loc de a o aplica intregii populatii
  - Folosirea unor operatori de variatie specifici
  - Modificarea selectiei pentru a preveni duplicarile
  - Modificarea criteriului de acceptare a unui vecin in cautarea locala
  - Adaptive Boltzman Operator (Krasnogor)

Exemplu: Boltzman MAs (Krasnogor)

- Foloseste un criteriu de acceptare inspirat Simulated Annealing: temperatura este invers proportionala cu diversitatea populatiei
- Problema de maximizare
- Δf = fitness vecin– fitness curent

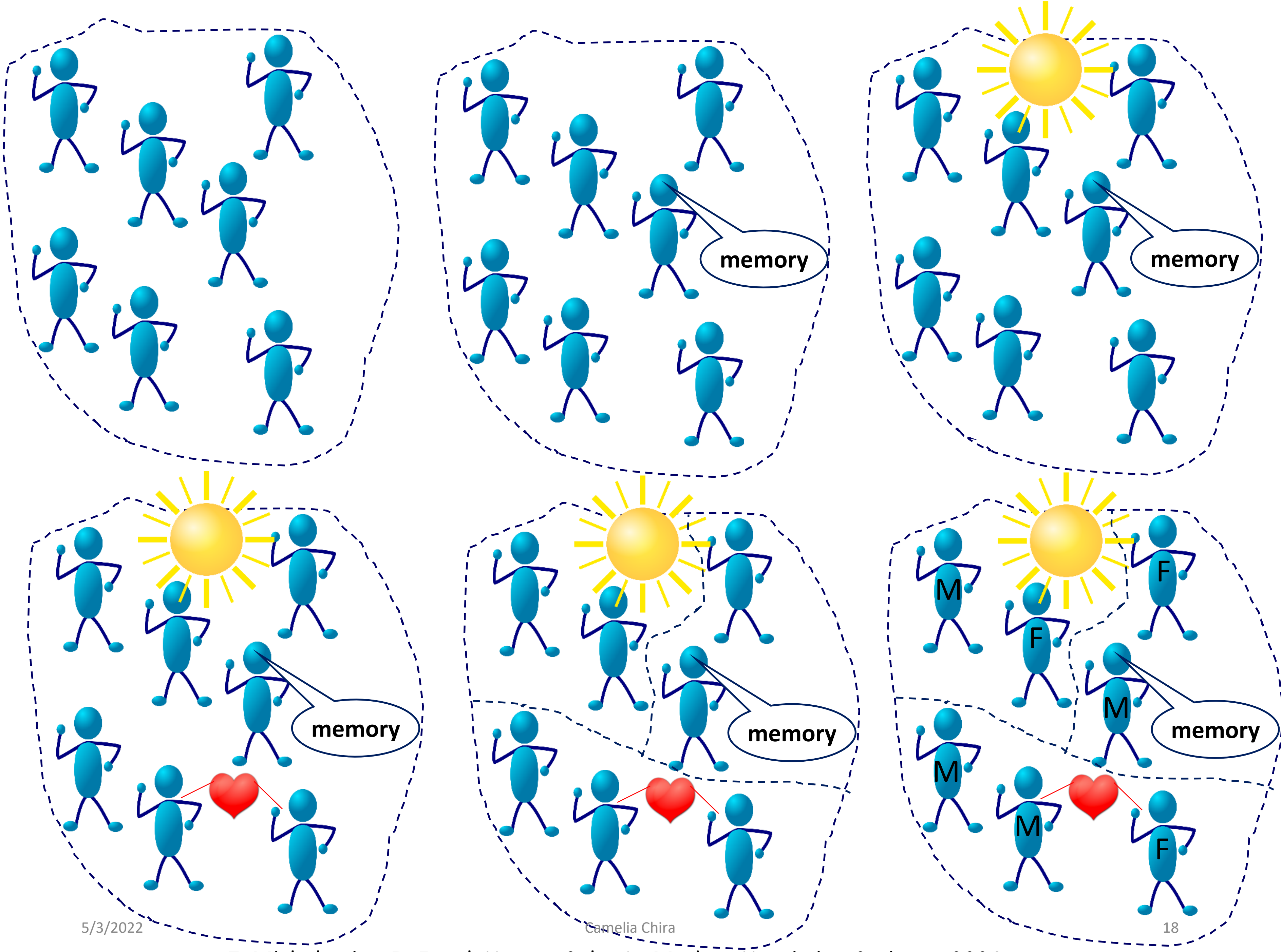
$$P(accepting\, neighbour) = \begin{cases} 1 & \Delta f > 0 \\ e^{\frac{k\Delta f}{f_{\max} - f_{avg}}} & \Delta f < 0 \end{cases}$$

Populatie diversa => fitness divers, *temperatura* mica, accepta numai vecini care imbunatesc fitness-ul => *Exploitation*

Populatia converge => *temperatura* mare, probabilitate mai mare de a accepta si vecini mai slabi => *Exploration*

Alagerea operatorilor in MA

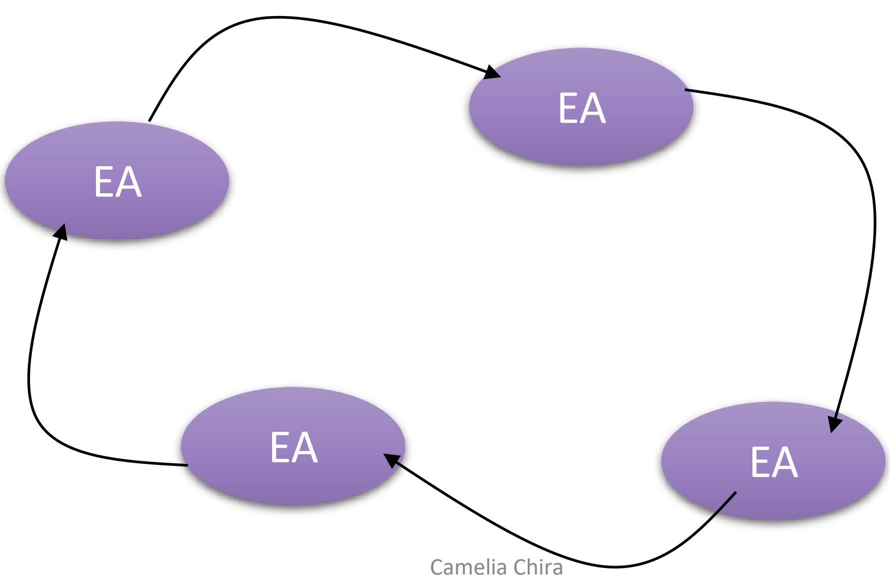
- Folosirea unui operator in cautare locala diferit de cel folosit pentru incrucisare si mutatie prezinta avantaje teoretice (Krasnogor, 2002)
- Folosirea mai multor operatori de cautare locala si stabilirea unui mecanism prin care sa se aleaga unul din ei
- Integrarea unor mecanisme de invatare si adaptare pentru alegerea unui operator
- Folosirea unor operatori din alti algoritmi folositi deja pentru problema abordata (ex. 2-opt pentru TSP)
- Integrarea unor cunostinte specifice problemei
- *Idei*
  - Folositi mai multi operatori de cautare locala in tandem
  - Adaugati o pozitie in individ care indica ce operator de cautare locala se foloseste (aceasta gena este mostenita de la parinti, si poate fi supusa mutatiei)



Z. Michalewicz, D. Fogel, How to Solve It: Modern Heuristics, Springer, 2004.

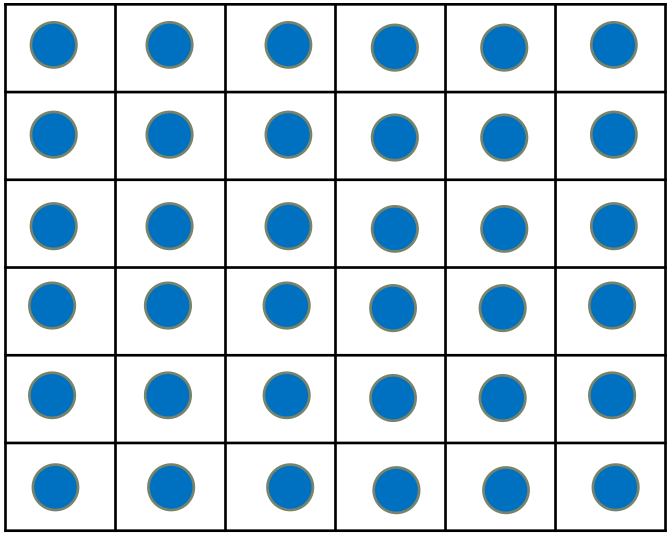
Posibilitati...ISLAND MODEL

- Putem imparti populatia in mai multe subpopulatii => **island, migration or coarse-grain models**
- Putem rula EAs diferiti pe fiecare subpopulatie (variatia si selectia la nivel de subpopulatie)
- Unul sau mai multi indivizi migreaza din cand in cand intre subpopulatii



Posibilitati...DIFFUSION MODEL

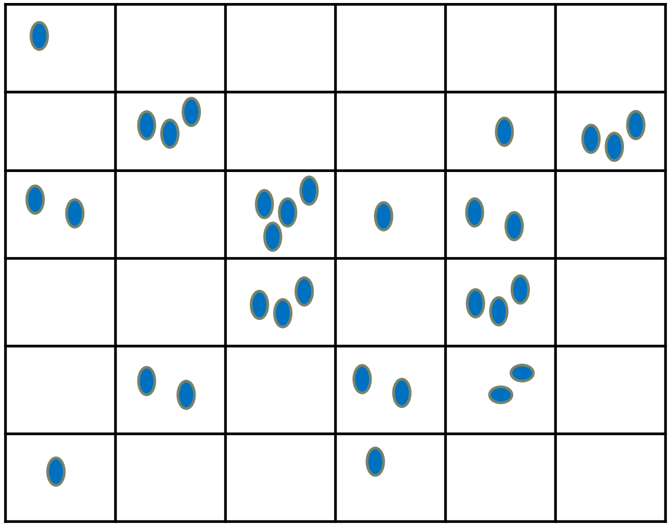
- Putem organiza populatia prin repartizarea fiecarui individ pe o zona geografica=> **diffusion, neighborhood or fine-grain models**
- Fiecare inidivid are o vecinatate data de topologie
- Selectia, variatia si inlocuirea parintilor sunt restrictionate de topologie

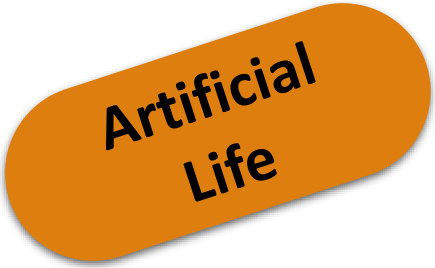


Topologie grid intr-un diffusion model clasic

Posibilitati...PATCHWORK MODEL

- Combina **island** and **diffusion models**
- **Patch** = fiecare celula din grid
- Fiecare inidivid este modelat ca un agent intr-o nisa care interactioneaza cu mediul si deciziile se bazeaza pe informatii locale
- Proprietati ale unui individ: naximum life span, ability to breed, mortality, preferences in decisions





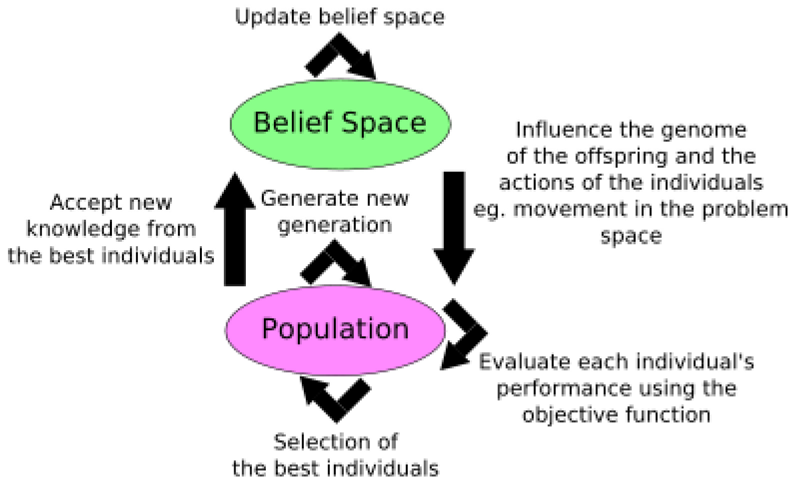
Cultural Algorithms

- Doua moduri de mostenire:
  - Microevolutionary level: trasaturi genetice si de comportament
  - Macroevolutionary level: credintele (beliefs) populatiei
  - Cele doua moduri interactioneaza printr-un canal de comunicatie care le permite indivizilor sa schimbe credintele si sa influenteze comportamentul individual pe baza credintelor populatiei

- *"evolution of evolution"*

(indivizii din populatie invata in timpul vietii)

  - Evolutie: invatare la nivel de specie, invatare sociala in societati umane



Sumar hibiridizari AE

- AEs ofera o abordare extrem de flexibila in rezolvarea problemelor
- Pot fi usor hibridizati cu algoritmi traditionali (ex. Metoda greedy in initializarea populatiei) sau pot fi extinsi pentru a include diferite aspecte observate in natura
- Idei: *populatii multiple, memorie, invatare individuala, invatare sociala, etc*

- **No Free Lunch Theorem**
  - Nici un algoritm nu este cel mai bun pentru orice problema.
  - *Indiferente de ce extensii si modificari i se aduc unui algoritm, vor fi intotdeauna probleme pentru care este foarte bun si altele pentru care merge extrem de rau*

Hibridizare SI models

- **Metaeuristica ACO**

procedure ***ACO***
while (not-termination-criterion)
schedule sub-procedures
generate-&-manage-ants()
update-pheromones()
execute-daemon-actions()
end schedule sub-procedures
end while
end procedure

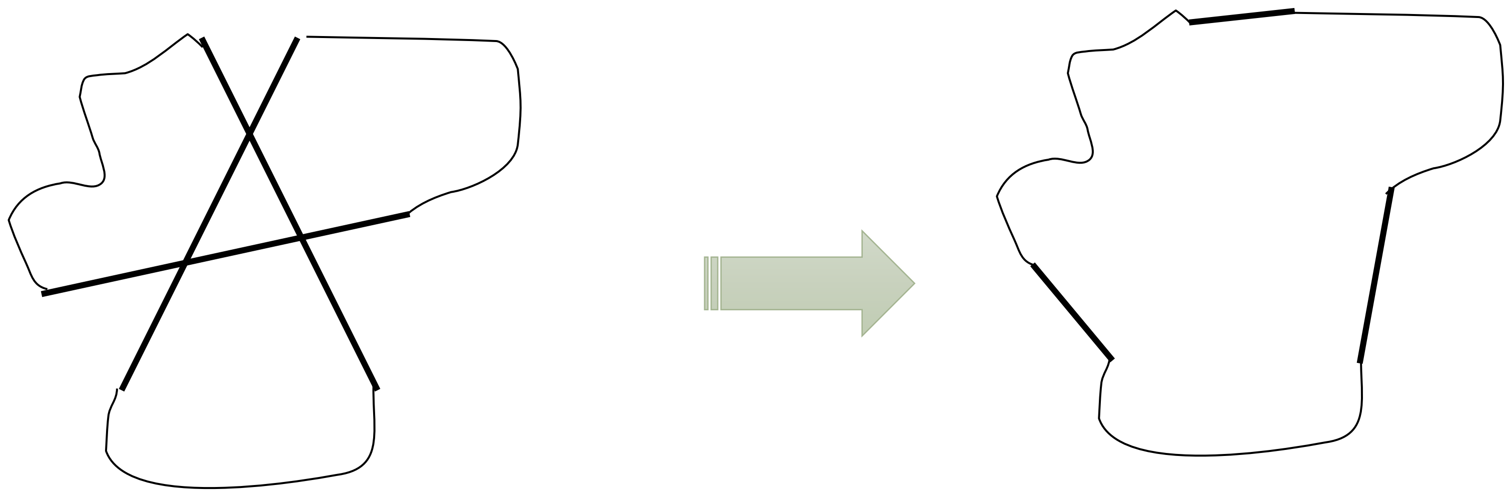
… + Local Search

ACS + Local Search

- **REPEAT**
  - Fiecare furnica este pozitionata intr-un nod de start
  - **REPEAT**
    - **State transition rule:** Fiecare furnica construieste incremental o solutie
    - **Local pheromone updating rule**
  - **UNTIL** Toate furnicile - un drum complet
  - Fiecare furnica - **Local Search (cautare locala)**
  - **Global pheromone updating rule:** Urmele de feromon sunt modificate la nivel global
- **UNTIL** STOP condition

ACS-3-opt = ACS + Cautare locala cu 3-opt

- La fiecare iteratie ACS, pentru fiecare furnica
  - 3 muchii odata sunt schimbate iterativ pana cand un optim local este atins (toate celelalte muchii sunt pastrate)
- Restricted 3-opt (pentru ATSP)
  - Ordinea in care orasele sunt vizitate nu se schimba
  - (k,l), (p,q), (r,s) -> (k,q), (p,s), (r,l)



ACS + cautare locala

- Cautarea locala este complementara mecanismului stigmergic
- Calitatea solutiilor este imbunatatita
- Cautarea locala porneste de la solutii semnificative generate de furnici
- Paralela cu algoritmi evolutivi:
  - Incrucisare – feromon
  - Mutatie – cautare locala
- Costul computational creste

Aplicatii

- **Optimizare numerica**

- **Probleme de optimizare combinatoriala**
  - Problema rucsacului
  - Traveling Salesman Problem
  - Vehicle Routing Problem
  - …
- **Alte probleme de optimizare**
  - Detectarea comunitatilor in retele complexe
  - Detectarea de reguli pentru automate celulare
  - Detectarea structurii proteinelor
  - Determinarea rutelor in trafic
  - …

Probleme de optimizare combinatoriala

- **Problema rucsacului (0/1 Knapsack Problem)**
  - Instante: https://people.sc.fsu.edu/~jburkardt/datasets/knapsack_01/knapsack_01.html
  - Reprezentare binara
  - Operatori de incrucisare si mutatie specifici codificarii binare
  - Functia de fitness: cu penalizare

- **Problema comis-voiajorului (Travelling Salesman Problem)**
  - Instante: http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/
  - Codificare prin permutari
  - Operatori de incrucisare si muatatie specifici permutarilor

Probleme de optimizare combinatoriala

- **Vehicle Routing Problem**
  - Instante: https://sites.google.com/site/vrphlibrary/benchmark-por

  - **Def**: given a fleet of vehicles with uniform capacity, a common depot, and several customer demands, finds the set of routes with overall minimum route cost which service all the demands
    - *Avem un depozit si mai multe orase, fiecare cu o anumita cerere*
    - *Avem un camion care are o anumita capacitate*
    - *Trebuie sa gasim rutele de cost minim de la depozit care sa viziteze toate orasele (tinand cont de restrictii)*

  - Reprezentare prin permutari:

    2 4 9 6 7 8 3 1 5
    - route1: [0 2 4 9 0]
    - route2: [0 6 7 8 3 0]
    - route3: [0 1 5 0]
  - Operatori de incrucisare si mutatie specifici

Alte probleme de optimizare

- **Detectarea comunitatilor in retele complexe**
(Community detection in complex networks)

- **Detectarea de reguli pentru automate celulare**
(Density classification in Cellular Automata)

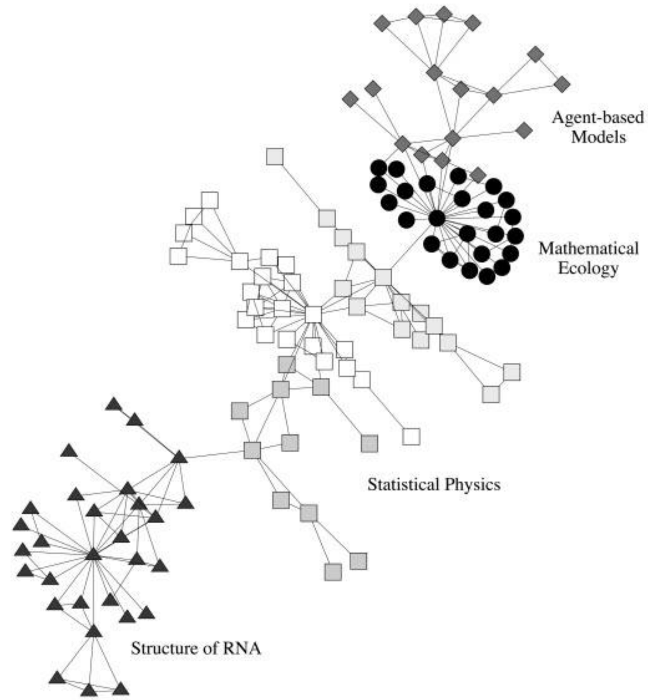
- **Detectarea structurii proteinelor**
(Protein structure prediction)

- **Probleme de optimizare a rutelor**
(Route Optimization)

- **Determinarea rutelor in trafic**
(Traffic Assignment Problem)

Complex Networks

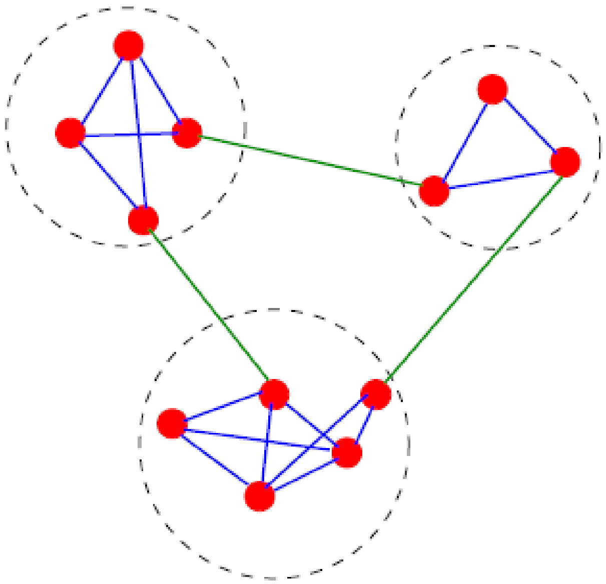
- ***Social networks***
  - acquaintance networks, collaboration networks
- ***Technological networks***
  - the Internet, the Worldwide Web, power grids
- ***Biological networks***
  - Neural networks, food webs, metabolic networks



- **Degree distribution**
  - typically many vertices in a network with low degree and a small number with high degree
- **Small-world effect**
  - the average distance between vertices in a network is short (13, 14)
- **Network transitivity**
  - two vertices both neighbors of the same third vertex have a high probability of being neighbors of one another

Community Detection Problem

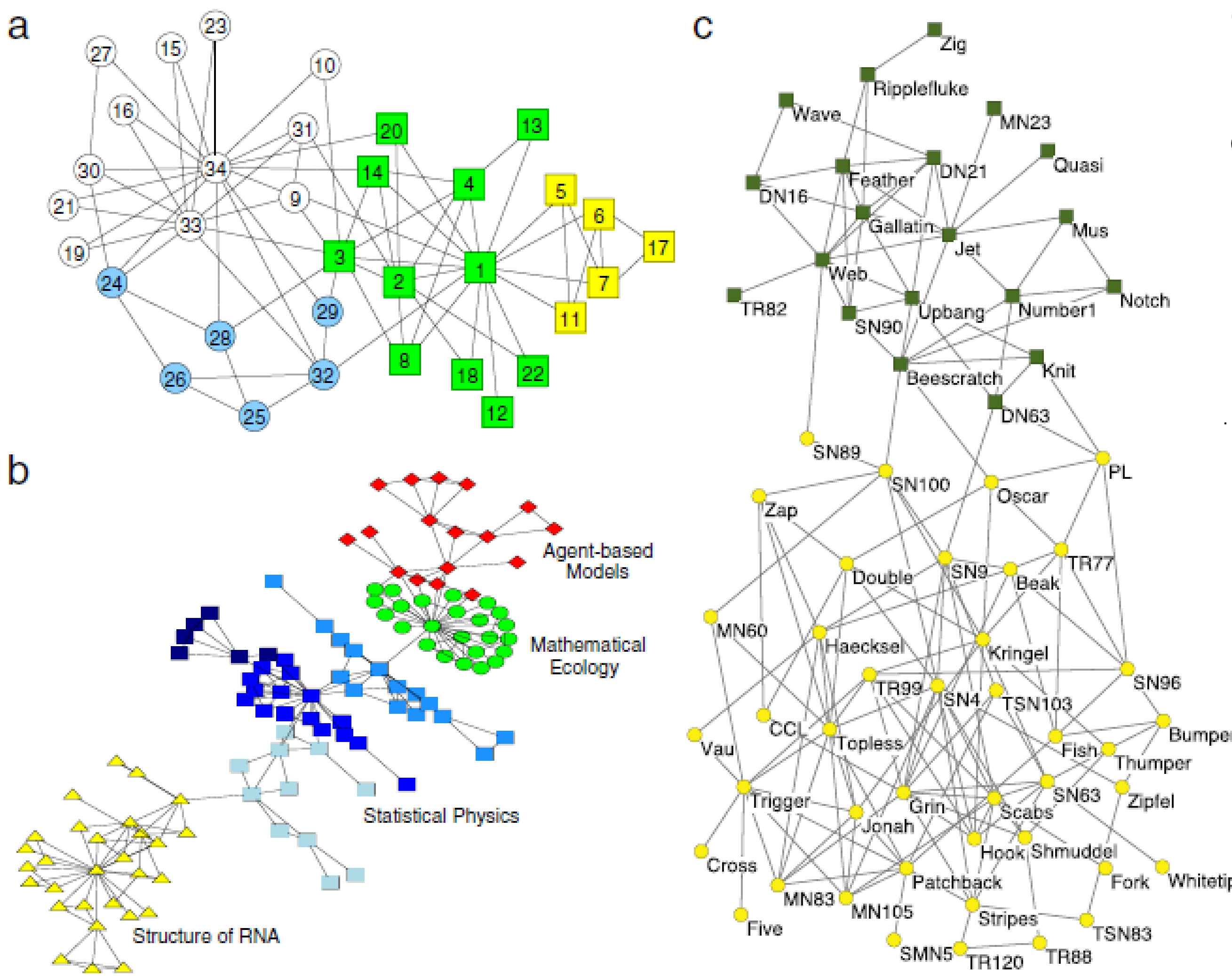
• A **community** in a network is a group of nodes densely connected but sparsely connected with the nodes belonging to other communities



Def. **The *community structure***

- *the division of networks into groups* (also called clusters) having dense intra-connections, and sparse inter-connections

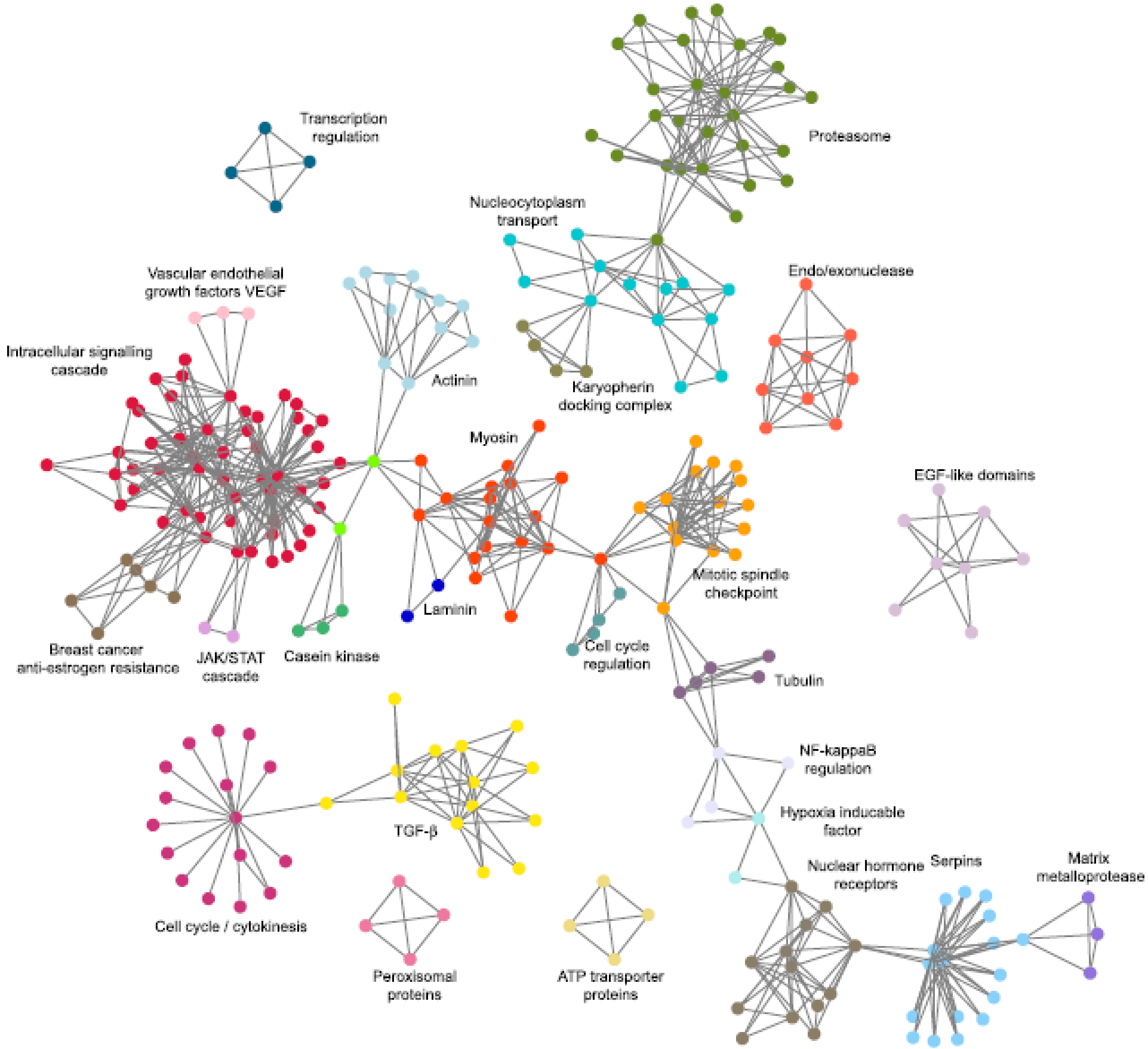
(a) Zachary's karate club, a standard benchmark in community detection



(c) Lusseau's network of bottlenose dolphins.

(b) Collaboration network between scientists working at the Santa Fe Institute.

S.Fortunato / Physics Reports 486 (2010) 75–174



S. Fortunato / Physics Reports 486 (2010) 75–174

Community structure in protein-protein interaction networks - the interactions between proteins in cancerous cells of a rat.

Definitions

- A social network: modeled as a graph G = (V,E)
  - where V is a set of objects, called nodes or vertices,
  - and E is a set of links, called edges, that connect two elements of V .
- **The adjacency matrix** (for N nodes):
  - the $N \times N$ *matrix* $\boldsymbol{A}$
  - *the entry at position (i, j) is 1 if there is* an edge from node *i to node j, 0 otherwise*

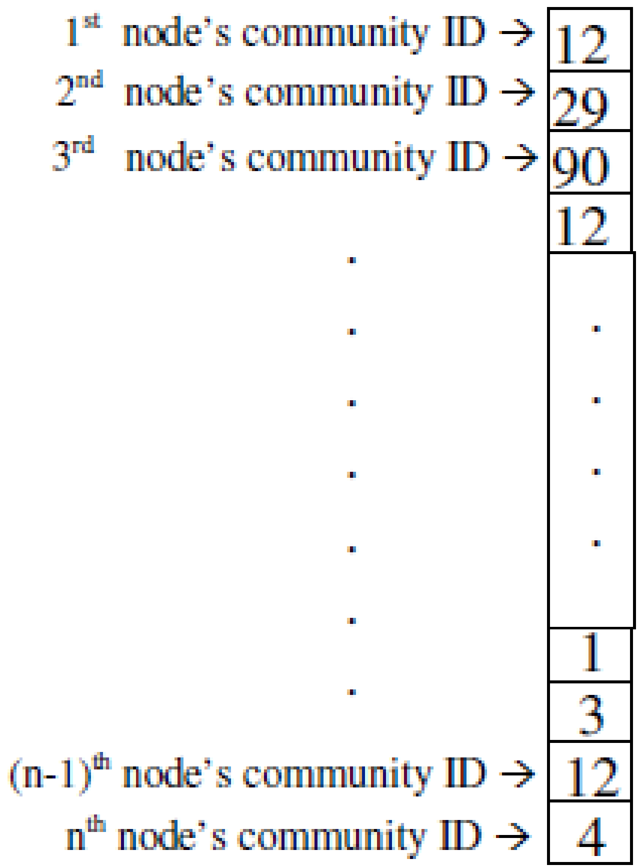
Community detection: Problem of detecting *k communities in a network :*

- Find a partitioning of the the nodes in *k subsets that are highly* intra-connected and sparsely inter-connected.

OR

- Find a partitioning of *A in k sub-matrices* that maximize the sum of densities of the sub-matrices.

Representation 1

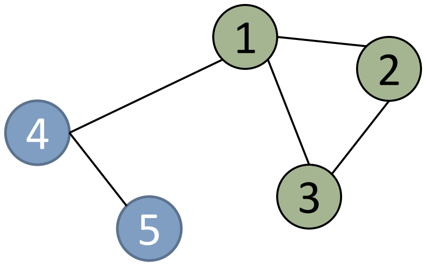


http://arxiv.org/ftp/cond-mat/papers/0604/0604419.pdf

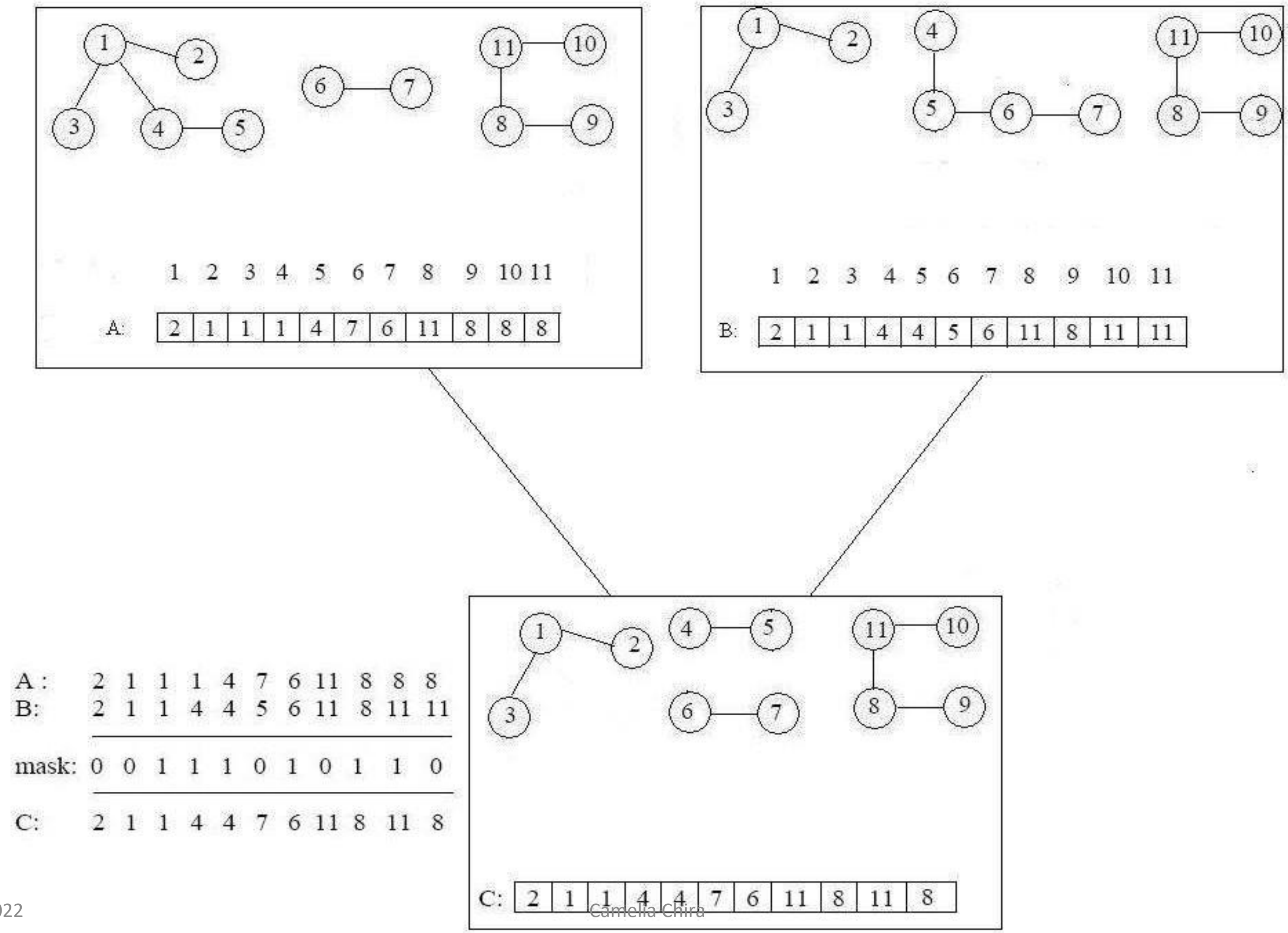
Representation 2

- Every individual is represented as an integer array of size *N*, where *N* is the number of nodes in the network
- Each position *i* in the array assumes a value *j* (where *j* can be an integer number from *1* to *N*) which is translated to a partitioning in which nodes *i* and *j* belong to the same cluster

1	2	3	4	5
3	1	2	5	4



Crossover



1 2 3 4 5 6 7 8 9 10 11
A: 2 1 1 1 4 7 6 11 8 8 8
1 2 3 4 5 6 7 8 9 10 11
B: 2 1 1 4 4 5 6 11 8 11 11
A : 2 1 1 1 4 7 6 11 8 8 8
B: 2 1 1 4 4 5 6 11 8 11 11
mask: 0 0 1 1 1 0 1 0 1 1 0
C: 2 1 1 4 4 7 6 11 8 11 8
C: 2 1 1 4 4 7 6 11 8 11 8

Fitness Function

- **Newman Modularity measure**

$$Q = \sum_i (e_{ii} - a_i^2)$$

- **C. Pizzuti, PPSN 2008**
  - *S = (I, J) sub-matrix of A*

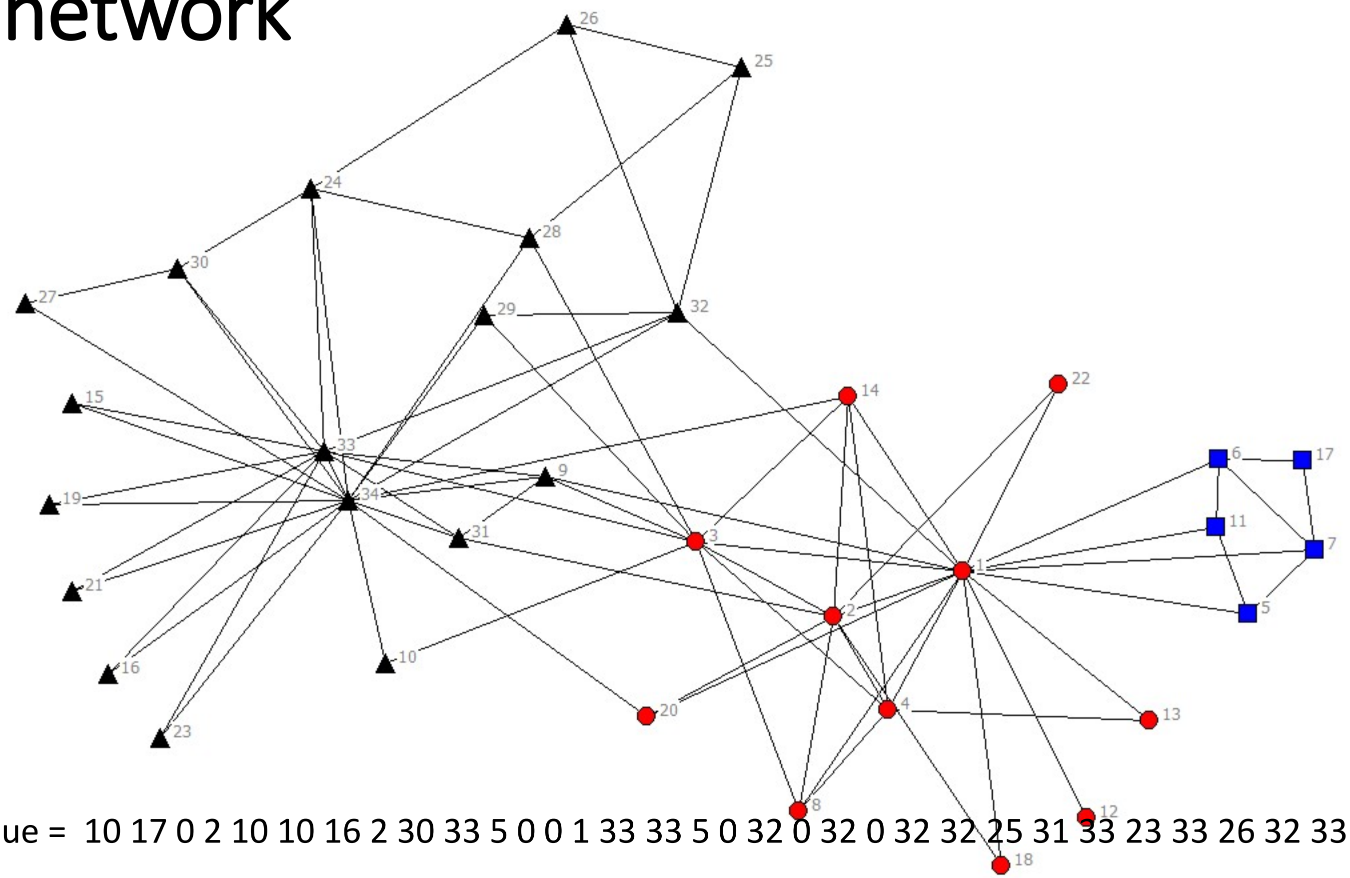
$$a_{iJ} = \frac{1}{|J|} \sum_{j \in J} a_{ij}, \text{ and } a_{Ij} = \frac{1}{|I|} \sum_{i \in I} a_{ij}$$

$$\mathrm{M}(S) = \frac{\sum_{i \in I} (a_{iJ})^r}{|I|}$$

$$\mathcal{CS} = \sum_i^k Q(S_i)$$

Community Score

Karate network

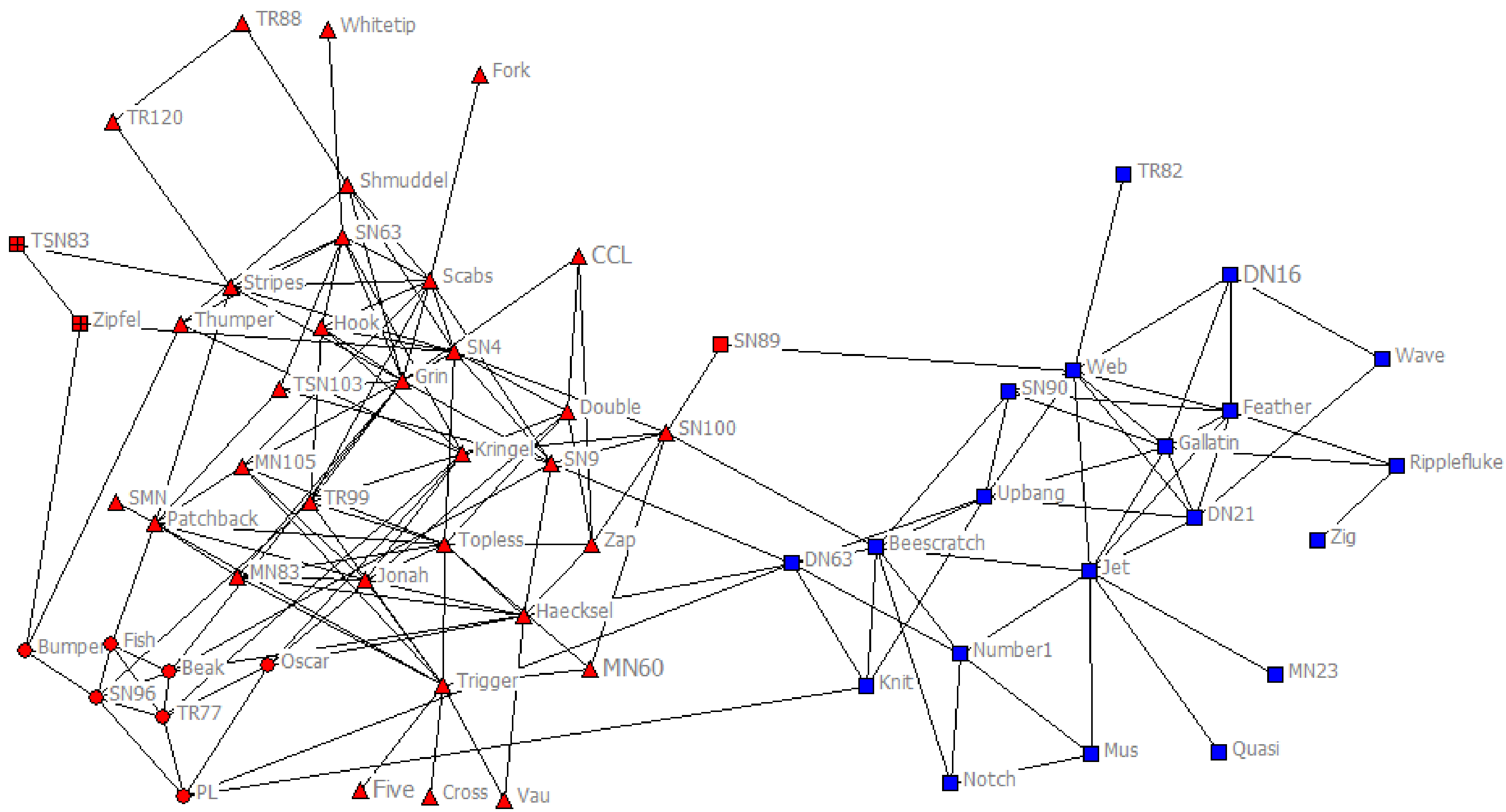


Chromosome value = 10 17 0 2 10 10 16 2 30 33 5 0 0 1 33 33 5 0 32 0 32 0 32 32 25 31 33 23 33 26 32 33 31 31

Cluster 0 : 1 3 4 8 12 13 20 22 2 18 14
Cluster 1 : 5 11 6 7 17
Cluster 2 : 9 31 33 32 19 21 23 24 28 10 34 15 16 27 29 30 25 26

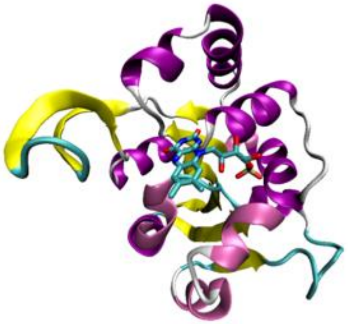
Newman modularity for global best: 0.39907955292570685
Normalized Mutual Information for global best: 0.8255181611085524

Community structure for dolphins network with NMI=1



The behavior of 62 dolphins over a period of 7 years. The number of edges is 159, each emphasizing a statistically significant frequent association.

Protein Folding



- The structure of a protein determines its function
  - The knowledge generated by a correct prediction of protein tertiary structures is of huge importance for a better treatment of certain diseases and can provide significant insights in the structure-based drug design field

- *"If a protein is to find its functional conformation space by wandering randomly through conformation space, an excess of 10^{50} years would be required for folding"* ***(Levinthal paradox)***

Unfolded chain of aminoacids

PF simulations

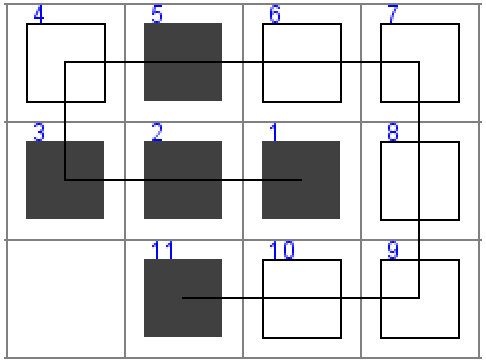
Protein structure – minimum energy

Simplified protein models

- The **hydrophobic-polar (HP)** model
  - The simplest - yet non-trivial - abstraction for the protein structure prediction problem
- A protein structure with ***n*** amino acids ⇔

  a sequence $\boldsymbol{S = s_1 \ldots s_n}$, $\mathbf{s_i}$, $i=1 \ldots n$ can be either ***H*** or ***P***

☑*A valid protein configuration for*
self-avoiding path on a regular latt
with vertices labeled by amino acid



S = HHHPHPPPPPH

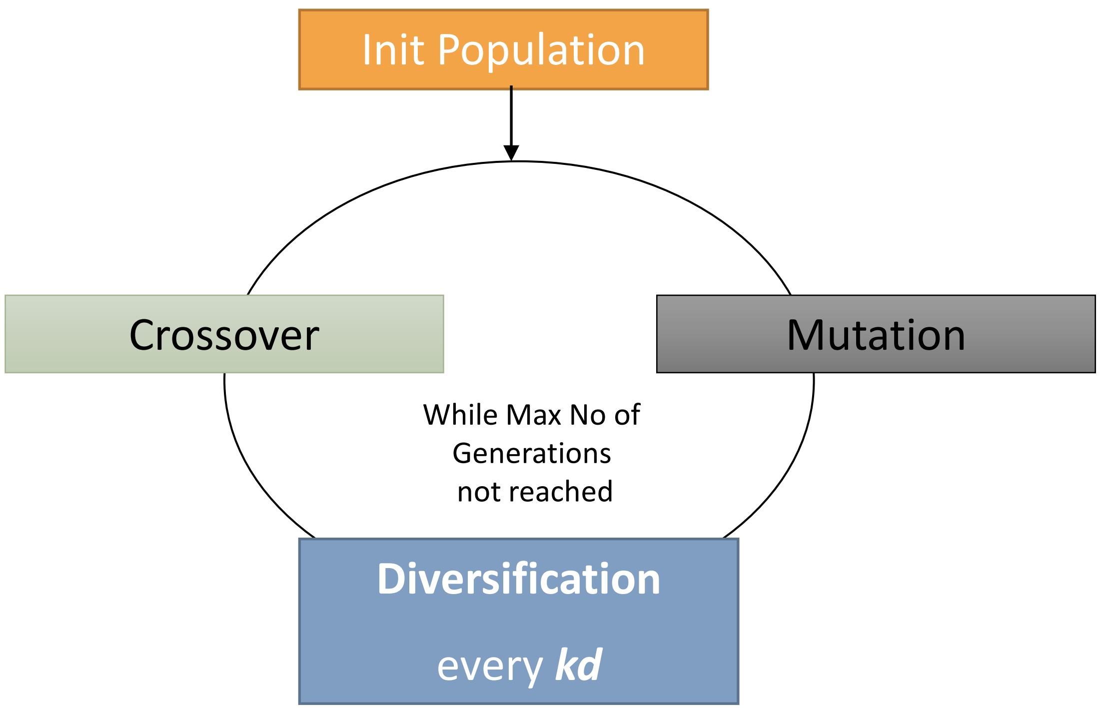
Protein Structure Prediction

- Elements of a given pair of residues are considered topological neighbors if they are adjacent in the lattice and not consecutive in the sequence
- The energy associated to a protein conformation takes into account every pair of H residues which are topological neighbors
- Every H-H topological contact contributes -1 to the energy function

- **Objective**: find the protein conformation with minimum energy

Evolutionary Model with Hill-Climbing Search Operators

- The population size is fixed and offspring are asynchronously inserted in the population

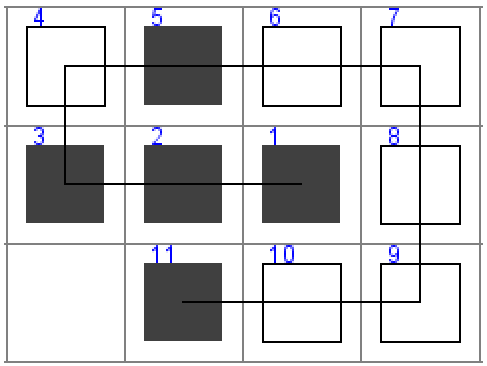


Representation

- Internal coordinates representation

- For a protein HP sequence with ***n*** residues ***S = s₁...sₙ*** , the chromosome length is ***n−1,*** each position in the chromosome encodes the direction
  - ➢L(Left)
  - ➢U(Up)
  - ➢R(Right)
  - ➢D(Down)

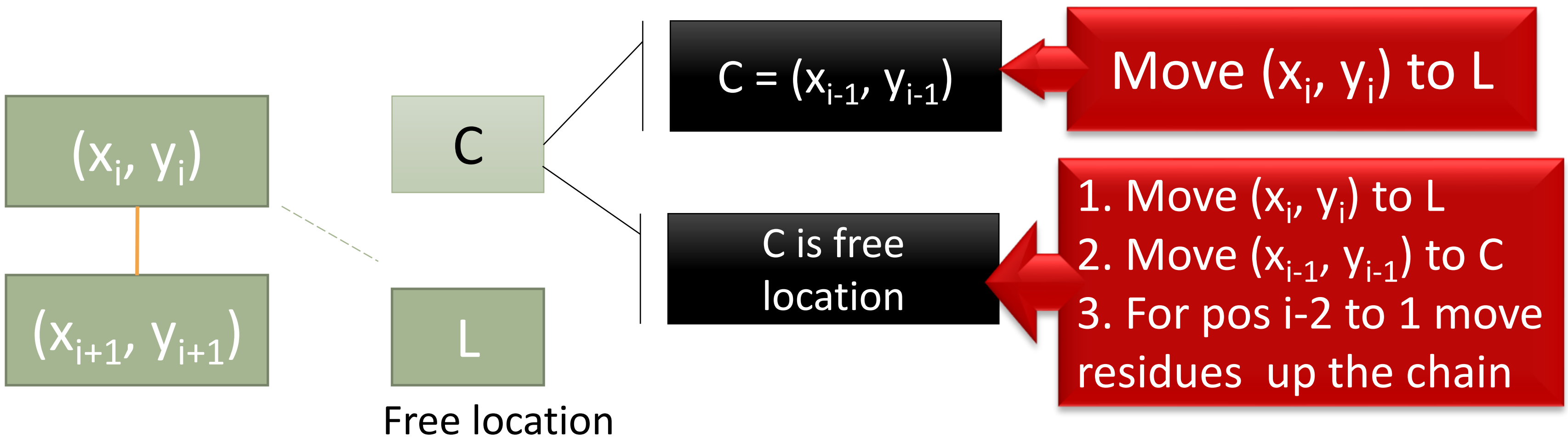
  towards the location of the current residue relative to the previous one.

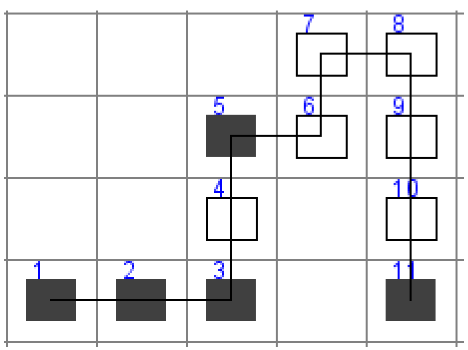


LLURRRDDLL

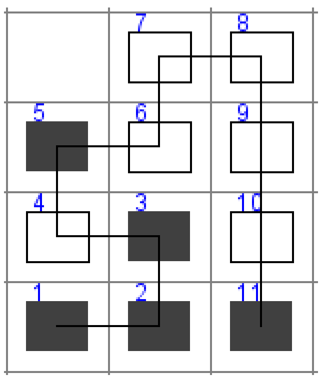
Mutation: Pull Moves (Lesh, 2003)

- A single residue is moved diagonally causing the transition of connecting residues



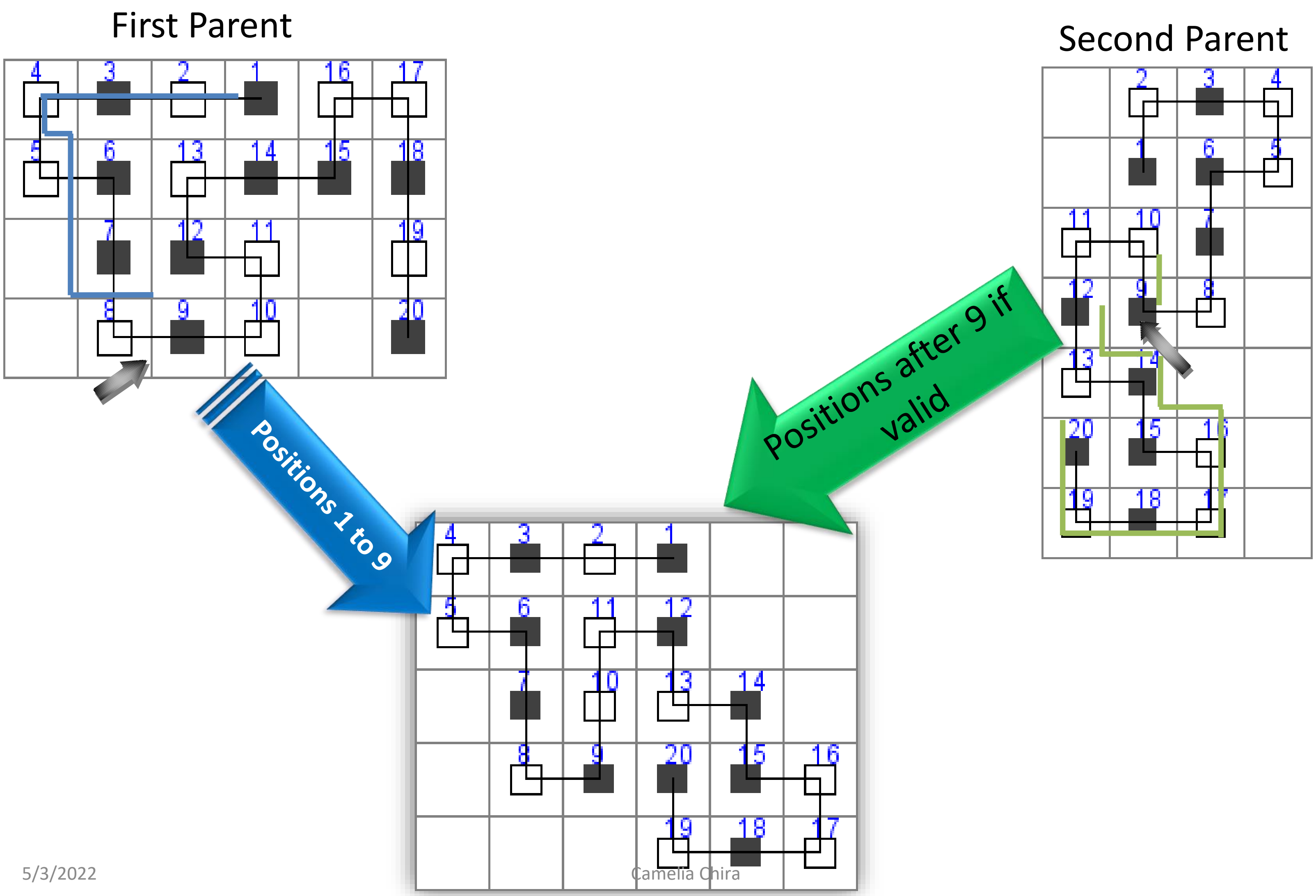


RRUURURDDD



RULURURDDD

Crossover Operator



Diversification

- Individuals from the current population are grouped based on their fitness (one group for each fitness value)

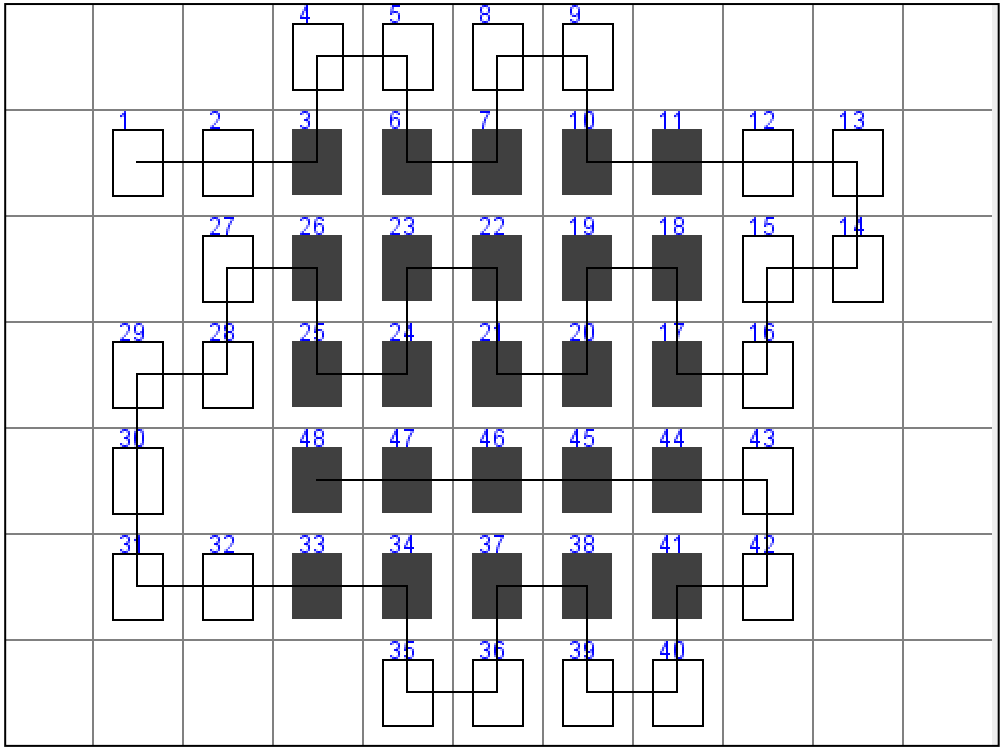
- For each group identified, subgroups of similar individuals are constructed based on the Hamming distance
  - Individuals are considered similar if $\boldsymbol{distH < (n-1)/4}$

- For each subgroup of similar individuals, one of them is kept in the current population and the rest of individuals are replaced by new randomly generated chromosomes (improved by a hill-climbing mutation).

Results for Seq. S5 size 48, E = -23

- *HP sequence*: 2P 1H 2P 2H 2P 2H 5P 10H 6P 2H 2P 2H 2P 1H 2P 5H

- *Chromosome*:
RRURDRURDRRRDLDLULDLULDLULDLDDRRRDRURDRURULLLLL



Routing Optimization

- ***Vehicle Routing Problem (VRP)***
  - *Capacitated VRP*
  - *Multiple Depots VRP*: 2 SC
  - *VRP with Pickup and Delivery*

- Route for a vehicle k: **R = {i_1, i_2,…,$i_{|R|}$}**, where $i_1 = i_{|R|} = -1$ or $i_1 = i_{|R|} = 0$
- R is feasible if:

  ✓ The number of bicycles required by BS in the route is not higher than the capacity of the vehicle $$\sum_{p=2}^{|R|-1}(d_{i_p} - s_{i_p}) \le Q_k$$

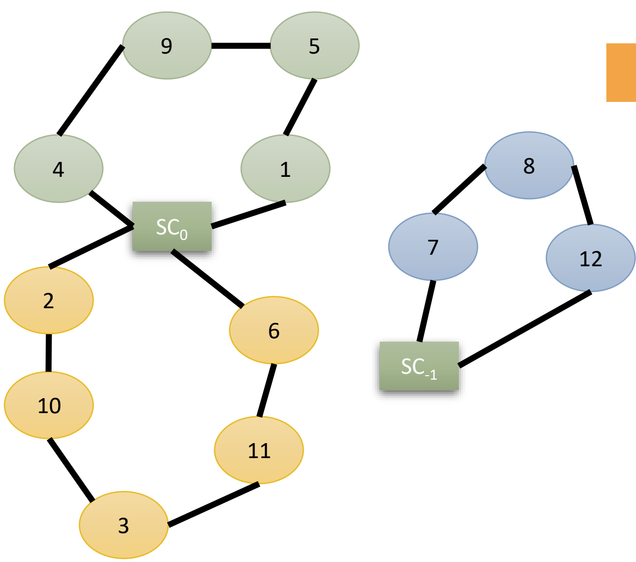
  ✓ For each BS: a positive demand should not exceed the current vehicle load y_k while a pickup demand should not exceed $Q_k - y_k$

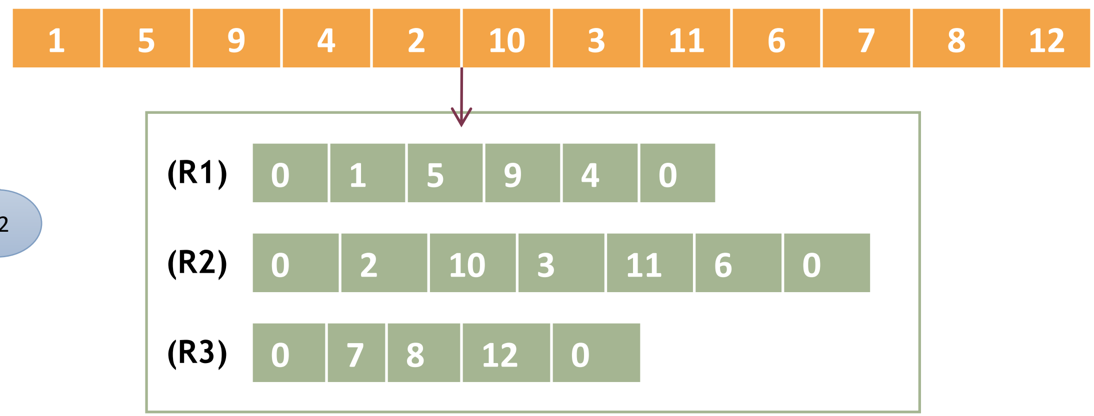
EA Features

- Fitness function: total cost of routes
- Fixed size population initially randomly generated
- Elite subpopulation + roulette-wheel selection
- Order crossover
- Swap mutation
- Asynchronous replacement of first parent: occurs during the same generation and the newly generated offspring might be selected as a mate within the same generation

Representation

- An individual incorporates all the routes to be executed and is defined as a permutation of the node
- The SC (nodes -1 and 0) are not included in the representation as they will be automatically determined based on the nearest SC to the first node in the route

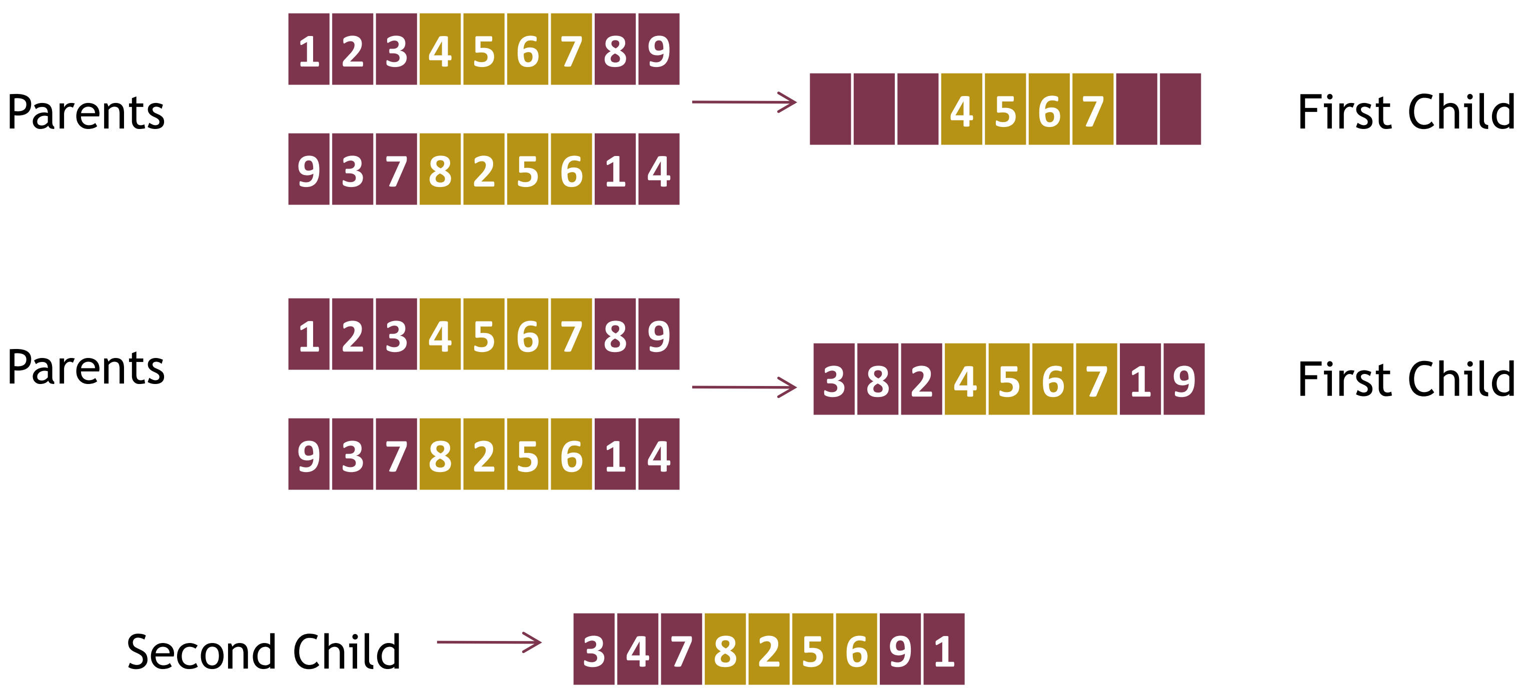




- Nodes are assigned to a route in the order they appear in the permutation
- *A new route is created when the* current node in the permutation can not be serviced by the same vehicle

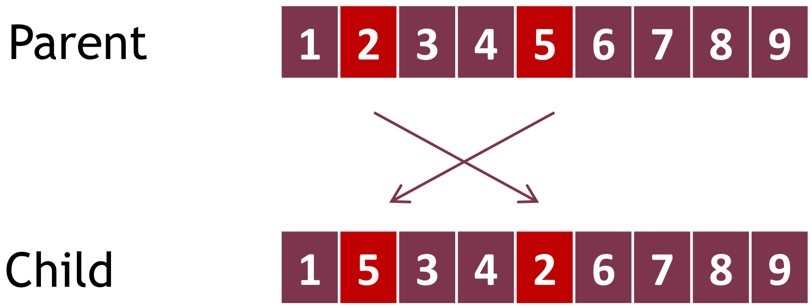
Order crossover

Order crossover is applied to this pair of parents with a certain probability.



Swap mutation

Swap mutation is applied to the current individual from the intermediary population with a certain probability.



Recapitulare

- Probleme complexe
  - Spatiul de cautare, Marimea problemei
- Cautare locala
  - Functii de vecinatate, Algoritmi Hill-Climbing
- Algoritmi single-solution
  - Simulated Annealing, Tabu Search
- Algoritmi Evolutivi
  - Reprezentare, fitness, selectie, variatie
  - Selectia si managementul populatiei, setarea parametrilor
- Algoritmi inspirati de natura (swarm intelligence)
  - PSO, ACO
- Invatare automata
- Modele hibride