Adaptive Spatial System Emergence from Community Swarm Optimization

Rawan Ghnemat(1),
Cyrille Bertelle(1) **& Gérard H.E. Duchamp**(2)

(1) LITIS - **Le Havre University - France**
(2) LIPN - **Paris XIII University - France**

SIAAS'08 within IASB 2008
April 3-4, 2008
Aberdeen, Scotland

Outline

1. Swarm Intelligence: scientific and historical context
2. Swarm Intelligence, Complexity and Emergence
3. Swarm Intelligence relevance for applications
4. A Community based appoach
5. Conclusion

Swarm Intelligence: scientific and historical context

Hardware history and computing concept impact

- Sequential architecture
  - sequential programming
- Parallel architecture
  - Splitting problem into predefined sub-problems
  - Master and Slaves model: centralized decomposition
- Distributed architecture
  - Decentralized approach based on processes interaction

Swarm Intelligence: scientific and historical context

Software history and problem solving concept impact

- Sequential programming and procedural programming
- Object-oriented programming
  - Operational entities
  - Private or public data and methods
  - Interfacing by public methods
  - Program consists in the objects interaction
- Agent-based programming
  - Autonomous entities within an environment
  - Include life cycle and eventually complex behavior
  - Interfacing by perceptions and actions
  - Program consists in virtual world simulation

Modelling history and simulating concept impact

- Numerical analysis computing
  - Equational approach: describes the global system behavior
  - Law-based description
  - Program consists in trajectories computing
  - Top-down approach
- Complex systems and distributed computing
  - Conceptual approach where entities are described by behavior
  - Rule-based description
  - Program consists in virtual world simulation
  - The global system behavior emerges from the entities interaction
  - Bottom-up approach

Outline

1. Swarm Intelligence: scientific and historical context
2. Swarm Intelligence, Complexity and Emergence
3. Swarm Intelligence relevance for applications
4. A Community based appoach
5. Conclusion

Swarm Intelligence, Complexity and Emergence

Swarm Intelligence is based on distributed solvers

- Not plannified sequential solvers
- Not splitting problem in many smaller ones and assemblying the sub-solutions
- But distributed solvers
  - which do not describe themselves the solution
  - but where the solution **emerges** from their interactions

Swarm Intelligence, Complexity and Emergence

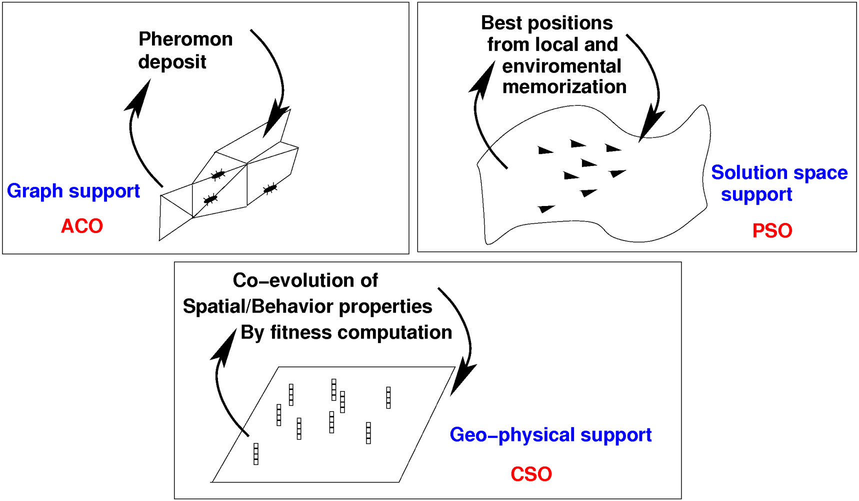
Emergence

- *weak* emergence: from a set of interacting entities, we can observe a system formation with some specific properties not described inside the entities themselves
- *strong* emergence: the emergent system feed-back on its own entities and can control them

Swarm Intelligence, Complexity and Emergence

Some swarm intelligence methods

- **Ant Colony optimization**: virtual ants moving on a graph which represents the contextual environment;
- **Particule swarm optimization**: virtual particles collectively moving on a solution space;
- **Community swarm optimization**: virtual automata population evolving and moving on spatial environment.


Pheromon deposit
Graph support
ACO
Best positions from local and enviromental memorization
Solution space support
PSO
Co−evolution of Spatial/Behavior properties By fitness computation
Geo−physical support
CSO

Outline

1. Swarm Intelligence: scientific and historical context
2. Swarm Intelligence, Complexity and Emergence
3. **Swarm Intelligence relevance for applications**
4. A Community based appoach
5. Conclusion

Swarm Intelligence relevance for applications

Optimization! ... and what else?

- When the optimization function could not be defined because the system and its properties are emerging
- The problem is not described by optimization but by adaptation

Swarm Intelligence relevance for applications

Geographical and social systems

- Urban dynamics
  - Gentrification: self-organization dynamic of city centers attraction and impact on the urban structure which will redefine its own centers by feed-back processes.
  - Mobility network: how the 3-D urban morphology emerge from the mobility interaction network and how these emergent morphologies feed-back on the interaction network?

Swarm Intelligence relevance for applications

Geographical and social systems

- Land-use modelling (Territorial intelligence)
  - Regional European Union help: What is the impact in respect of the interaction network (economical, environmental, ...) inside and outside the target region to be helped.
- Sustainable development
  - Complex network of objectives: economical, social and environmental
  - Multi-scale in space and time, according to each objective

Outline

1. Swarm Intelligence: scientific and historical context
2. Swarm Intelligence, Complexity and Emergence
3. Swarm Intelligence relevance for applications
4. **A Community based appoach**
5. Conclusion

A Community based appoach

Community Definition

- **Definition**: A *community* is a system or an organization which is characterized by a *spatial* property, a *behavior* property and the interaction between the both.
- **Examples**:
  - Community in Ecology (food chains);
  - Schelling's segregation model patterns are the results of such emergent communities, based on elementary behaviors.

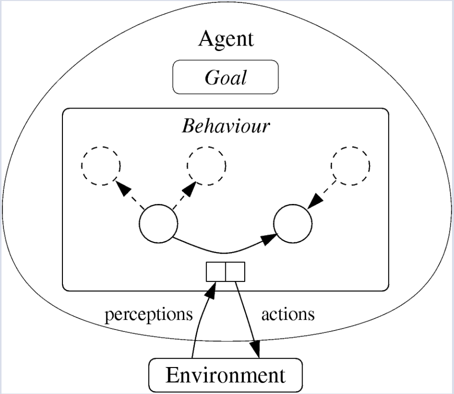
Community swarm optimization

Principles

- Entities describing rule-based transitions systems (agent behaviors) and implemented by automata with multiplicities ...
- ... on which we can defined behavioral distance allowing emergence of organizations
- Implementation of the system feed-back (adaptation or control) using genetic processes ...
- ... and leading to emergent communities, mixing spatial and behavioral properties.

Spatial Behavioral Automata

Automata-based Agent Behavior



Spatial Behavioral Automata

Automata with multiplicities

Agent behavior is modelled by automata with multiplicities which is defined by

- A set of perception represented by an alphabet
- A set of actions represented by a **semi-ring** K
- A set of states with a subset of initial states and a subset of final states
- A set of transitions between states which is generate by a perception in input and which generate an action in output

Spatial Behavioral Automata

Automata with multiplicities

Because the set of actions K is a **semi-ring**,

- we can represented the automata using a linear representation (vectors and matrices),
- we can defined many kinds of operators on these automata and so improve automatic processes on agent management.

Spatial Behavioral Automata: example

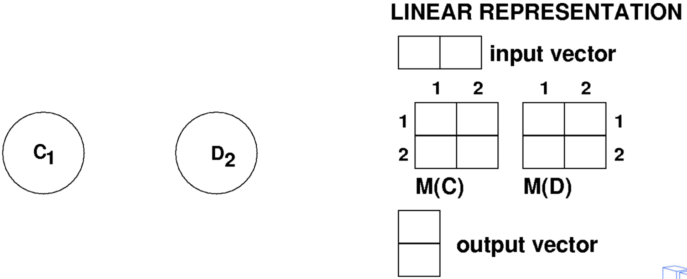
- Strategy modeling using *probabilistic* automata for game theory
- Automata based model for player behavior with adversory
  - 2 behavioral states: Cooperate (s1: C) or Defect (s2:D)

Spatial Behavioral Automata: example

- Strategy modeling using *probabilistic* automata for game theory
- Automata based model for player behavior with adversory
  - 2 behavioral states: Cooperate (s1: C) or Defect (s2:D)
  - Probabilistic transition from one state to another according to what make the adversory at the previous step

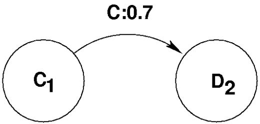
Spatial Behavioral Automata: example

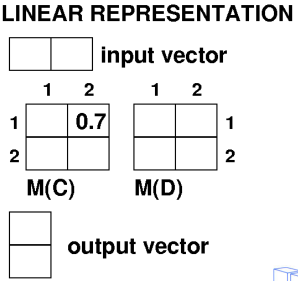
- Strategy modeling using *probabilistic* automata for game theory
- Automata based model for player behavior with adversory
  - 2 behavioral states: Cooperate (s1: C) or Defect (s2:D)
  - Probabilistic transition from one state to another according to what make the adversory at the previous step



Spatial Behavioral Automata: example

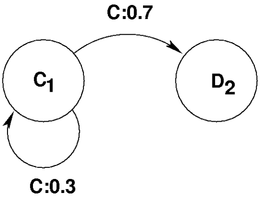
- Strategy modeling using *probabilistic* automata for game theory
- Automata based model for player behavior with adversory
  - 2 behavioral states: Cooperate (s1: C) or Defect (s2:D)
  - Probabilistic transition from one state to another according to what make the adversory at the previous step

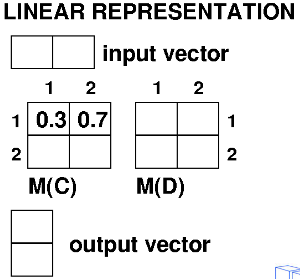




Spatial Behavioral Automata: example

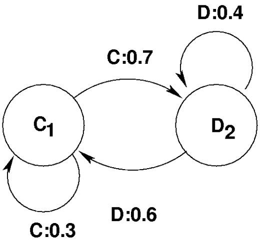
- Strategy modeling using *probabilistic* automata for game theory
- Automata based model for player behavior with adversory
  - 2 behavioral states: Cooperate (s1: C) or Defect (s2:D)
  - Probabilistic transition from one state to another according to what make the adversory at the previous step

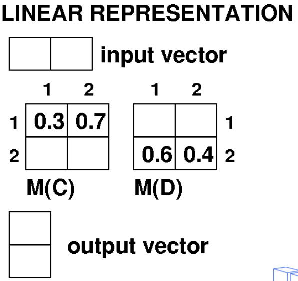




Spatial Behavioral Automata: example

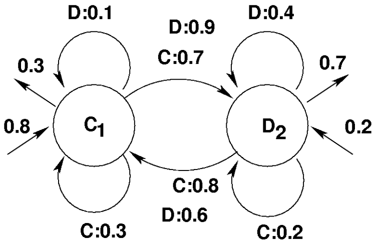
- Strategy modeling using *probabilistic* automata for game theory
- Automata based model for player behavior with adversory
  - 2 behavioral states: Cooperate (s1: C) or Defect (s2:D)
  - Probabilistic transition from one state to another according to what make the adversory at the previous step





Spatial Behavioral Automata: example

- Strategy modeling using *probabilistic* automata for game theory
- Automata based model for player behavior with adversory
  - 2 behavioral states: Cooperate (s1: C) or Defect (s2:D)
  - Probabilistic transition from one state to another according to what make the adversory at the previous step



LINEAR REPRESENTATION

0.8	0.2	input vector

M(C):

	1	2
1	0.3	0.7
2	0.8	0.2

M(D):

1	2	
0.1	0.9	1
0.6	0.4	2

output vector:

0.3
0.7

Spatial Behavioral Automata

Spatial automata and associated distances

- A spatial agent is defined by
  - Spatial coordinates
  - A behavior modeled by an automaton with multiplicities
- A **spatial distance** between 2 agents, can be computed according to their spatial coordinates
- A **behavioral distance** between 2 agents can be compute by the distance between the vectors which stores all the coefficients of the linear representation of the agent behavior automata.

Spatial Behavioral Genetic Automata

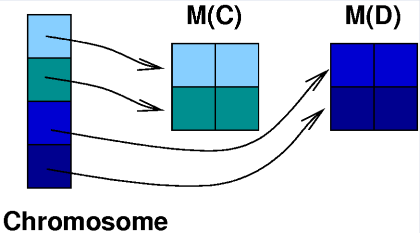
Genetic operators on automata population

- Genetic operators deal with **population** of **individuals** as spatial behavioral automata.
- **Individual** is described by a **chromosome** which is a sequence of **alleles** (elementary information).
- Here, the chromosomes are coding the transition matrices of the behavioral automata linear representation.
- Here, an allele is a matrix line ...

Spatial Behavioral Genetic Automata

Genetic operators on automata population

- ... and the chromosome is the set of the matrix lines of all the transition matrices



Community Swarm Optimization Algorithm

Overview

- CSO Algorithm consists in generating an initial virtual automata population describing some spatial transition rules system;
- This virtual automata population evolves and moves on a spatial environment;
- The evolution (and the moving) follows a genetic algorithm including a selection process associated to a fitness function.

Community Swarm Optimization Algorithm

Community Detection associated to fitness function

- We can defined the fitness of each agent as following:
  - We compute his neighbourhood, using the ***spatial distance***
  - We sum the ***behavioral distance*** of the agent itself with all the others agents included in the neighbourhood
  - We define the fitness as the inverse of the average of the previous sum.
- Self-organization communities emerge from the use of this fitness inside a genetic algorithm.

Community Swarm Optimization Algorithm

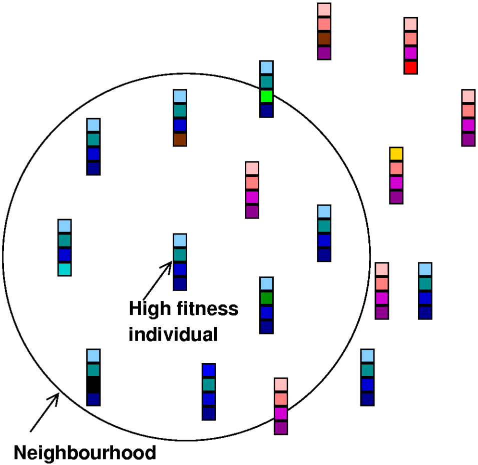
Community Detection associated to fitness function

Let $\mathcal{V}_x$ a neighbourhood of the agent x, relatively to its spatial location. We define $f(x)$ the agent fitness of the agent x as :

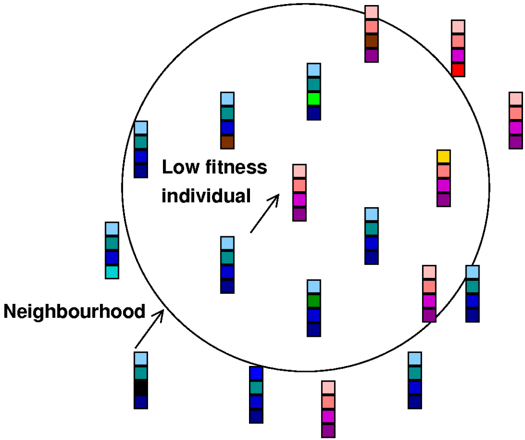
$$f(x) = \begin{cases} \dfrac{card(\mathcal{V}_x)}{\sum\limits_{y_i \in \mathcal{V}_x} d(x, y_i)^2} & \text{if } \sum\limits_{y_i \in \mathcal{V}_x} d(x, y_i)^2 \neq 0 \\ \infty & \text{otherwise} \end{cases}$$

where $d(x, y)$ is the behavioral semi-distance between the two agents x and y.

CSO Algorithm: example ... following



CSO Algorithm: example ... following



Outline

1. Swarm Intelligence: scientific and historical context
2. Swarm Intelligence, Complexity and Emergence
3. Swarm Intelligence relevance for applications
4. A Community based appoach
5. **Conclusion**

Conclusion and Perspectives

CSO specificity

- With the comparison of other methods from this category (ACO and PSO), CSO differs mainly on the modelling purpose;
- CSO deals with transition rules included data structures (automata) and algebraic operators allowing to implement automatic computation for self-organization.

Conclusion and Perspectives

CSO Applications

- Presented here as a generic method, CSO is efficient in engineering problems where spatial characteristics are not only additional coordinates for the data but contribute to self-organization
- Urban and land usage management need this kind of modelling to improve decision making.