Emergence [Wikipedia]

In philosophy, systems theory, science, and art, emergence occurs when a complex entity has properties or behaviors that its parts do not have on their own, and emerge only when they interact in a wider whole.

Emergence plays

Flock and Schools

Socio-cognitive Underpinnings

Universal individual behavior principles:

Evaluate tendency to evaluate stimuli (positive/negative, attractive/repulsive) is shared among all kinds of living organisms

Compare Comparing to others is a driver for learning and change

Imitate Effective strategy for learning, through not many living organisms are capable of full imitation

Rules

Emergent behavior in flocks and schools can be reduced to simple rules:

Separation an individual should avoid crowding or colliding with its neighbors

Alignment an individual should steer in the average heading of its neighbor

Cohesion

Particle Swarm Optimization (PSO)

In computational science, particle swarm optimization (PSO) is a computational method that optimizes a problem by iteratively

Particle Properties

Current position

Best position

Velocity

Parameters

- Number of particles K
- c_1 significance of personal particle experience (trust in individual knowledge)
- c_2 significance of swarm experience (trust in social knowledge)
- Inertia weight w
  - Sort of like a learning rate
- $V_{\max}$ velocity cap
- N_i neighborhood of particle i

Cycle

Create a swarm (population) of K particles initialized with:

- random position (point) in search space
  - best if uniformly distributed over space
- velocity set to:

  - zero or
  - random in $[-V_{\max}, V_{\max}]$

- Use fitness function to determine how fit (how well it solves the problem) each particle is initially.
- Update particle position (point in search space) and velocity while balancing exploitation and exploration.

Where Next?

- $x^t_{G,\ best}$=swarm's (global) best position
- $x^t_{i,\ best}$=Particle i's best position

Particle Position Update

Next position:

$$x_i^{t+1} = x_i^t + \vec{v}_i^{t+1}$$

where

$\mathbf{x_i^{t+1}}$ i's next position

$\mathbf{x_i^t}$ i's current position

$\vec{\mathbf{v}}_\mathbf{i}^\mathbf{t+1}$ i's next velocity

Next velocity:

$$\vec{v}_i^{t+1} = w \times \vec{v}_i^t + c_1 \times a \times (x^t_{i,\ best} - x_i^t) + c_2 \times b \times (x^t_{G,\ best} - x_i^t)$$

where

$a =$ random number in $[0; 1]$

$b =$ random number in $[0; 1]$

Also common:

$$\vec{v}_i^{t+1} = \vec{v}_i^t + c_1 \times a \times (x^t_{i,\ best} - x_i^t) + c_2 \times b \times (x^t_{G,\ best} - x_i^t)$$

Psychosocial Compromise

Check the termination condition

Controlling the Search

- If velocity is too low → algorithm too slow
- If velocity is too → algorithm too unstable

c_1 and c_2 Parameters

Should add up to 4.

$V_{\max}$ Velocity Cap

To better control particle trajectory and prevent stochastic velocity change to have uncontrolled

Diversification

Particles need time to

Characteristics

PSO has a memory:

- "where" the best solution was (as opposed to "what")

Quality population respond

Advantages / Disadvantages

Advantages

- Insensitive to scaling of design variables
- Simple implementation
- Easily parallelized

Variants / Modifications

- 2-D Otsu PSO
- Active Target PSO
- Adaptive PSO

Heuristics and Metaheuristics

Heuristics

- how to choose the next neighbor?
- use local information (state and its neighborhood)
- direct the search towards a local maximum

Other Swarm Algorithms

- Bat Algorithm
- Artificial Fish Swarm
- Cuckoo Swarm

Gradient Descent