# Swarm Intelligence

## Other ACO Algorithms and ACOTSP

Leonardo Bezerra and Leslie Perez Caceres

IRIDIA – Université Libre de Bruxelles (ULB)
Bruxelles, Belgium
lperez@iridia.ulb.ac.be
leonardo@iridia.ulb.ac.be

# Outline

1. Short Overview
2. ACOTSP
    1. Elitist Ant System
    2. Ranked-based Ant System
    3. Best-worst Ant System
3. ACOTSP options
    1. Algorithms
    2. Local search

# Review

- We have studied:
  - Ant System
  - Max-Min Ant System
  - Ant Colony System

- What they have in common?
  - Each ant builds a solution in each iteration
  - Ants are guided by pheromone (+ heuristic information)
  - Pheromone evaporates during the search
  - Pheromone is updated by ants according to the solution founded

# Review
## Ant System

```
1 For #iterations
2   For each ant Do
3     Choose a random starting city
4     While tour is not complete
5       Select next city [random proportional transition rule]
6     EndWhile
7   EndFor
8   Update pheromone [all ants,
                      tour length proportional]
9 EndFor
```

# Review
## Max-min Ant System

```
 1 For #iterations
 2   For each ant Do
 3     Choose a random starting city
 4     While tour is not complete
 5       Select next city [random proportional transition rule]
 6     EndWhile
 7   EndFor
 8   Update pheromone [(best) ant,
                       tour length proportional,
                       respect max and min pheromone values]
 9   If convergence Then
10     Reinitialize pheromone
11   EndIf
12 EndFor
```

# Review
## Ant Colony System

```
 1 For #iterations
 2   For each ant Do
 3     Choose a random starting city
 4     While tour is not complete
 5       Select next city [pseudo-random proportional transition rule (q0)]
 6       Local Update of pheromone
 7     EndWhile
 8   EndFor
 9   Global Update of pheromone [global best ant,
                                 tour length proportional]
10 EndFor
```

# ACOTSP

- ACOTSP developed by Thomas Stutzle, provides the implementation of a set of ACO algorithms to solve TSP.
- Which algorithms are implemented?
  - **Ant System**
  - Elitist Ant System
  - **Max-min Ant System**
  - Rank based Ant System
  - Best-worst Ant System
  - **Ant Colony System**

# ACOTSP
## Elitists Ant System

- Elitism refers to favour best individuals to guide the search. → intensification
- After each iteration the **global best ant** deposit pheromone along with the others.
- Introduce a new parameter ***e*** that controls the contribution of the global best ant to the pheromone update.

$$\tau_{ij}(t) = (1-\rho)\cdot\tau_{ij}(t-1) + \sum_{k=1}^{m} \Delta\,\tau_{ij}^{k} + e\,\Delta\,\tau_{ij}^{bs}$$

# ACOTSP
## Rank-based Ant System

- A number of the best ants are allowed to update pheromone.
- All the ants are **ranked** regarding their tour quality and the best ω-1 are selected.
- They deposit pheromone according to their rank. So the best ones contribute more.
- Parameter ω controls the **number of ants** allowed to deposit pheromone (usually 25%) and also controls the **amount of pheromone** contributed by each ant.
- The global best ant deposit pheromone with the others.

$$\tau_{ij}(t) = (1-\rho)\cdot\tau_{ij}(t-1) + \sum_{r=1}^{\omega-1}(\omega-r)\,\Delta\,\tau_{ij}^{r} + \omega\,\Delta\,\tau_{ij}^{bs}$$

# ACOTSP
## Best-worst Ant System

- Transition rule and pheromone evaporation as in Ant System
- Pheromone update after each iteration:
  - The global best ant contributes positively to the pheromone update
  - The worst ant contributes negatively to the pheromone update (additional evaporation)
    - This is only applied in the edges present in the worst ant and absent in the global best ant.
- Pheromone trails **mutation** → diversification
- Restart of the search when stagnation ($\tau_0$)

# ACOTSP
## Options: Algorithms

- How to specify the algorithm?
  - ***--as*** *: Ant System*
  - ***--eas*** *: Elitist Ant System*
  - ***--ras*** *: Rank-based version of Ant System*
  - ***--mmas*** *: MAX-MIN ant system*
  - ***--bwas*** *: Best-worst ant system*
  - ***--acs*** *: Ant colony system*
- Look for other parameters using ***./acotsp –help***
- Related parameters:

  ***--q0:*** q_0: prob. of best choice in tour construction (ACS)

  ***--elitistants:*** number of elitist ants (MMAS)

  ***--rasranks:*** number of ranks in rank-based Ant System (RAS)

# ACOTSP
## Options: Other

- Other general parameters

***--tries***:  number of independent trials (runs)

***--tours***: number of steps in each trial (max tours evaluated per trial)

***--time***: maximum time for each trial (seconds)

***--seed***: seed for the random number generator

***--optimum***: to stop if tour better or equal optimum is found

***--ants***: number of ants

***--nnants***: nearest neighbours in tour construction

- To use of candidate list to construct solutions

***--alpha***: alpha (influence of pheromone trails)

***--beta***: beta (influence of heuristic information)

***--rho***: rho (pheromone trail evaporation)

**--localsearch**:  0: no local search  1: 2-opt  2: 2.5-opt  3: 3-opt

# ACOTSP

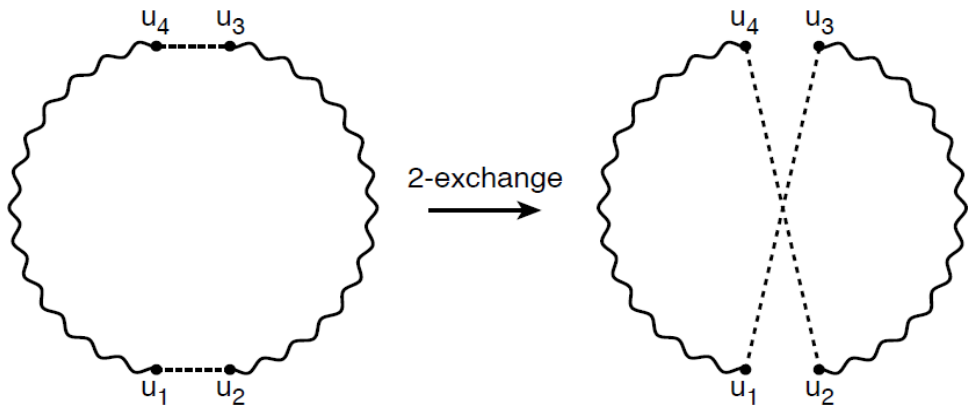
## Options: Local search

- **Local search** starts from a solution already constructed and moves through the search space from one neighbour to other.
- ACOTSP offers the possibility of apply a local search procedure to improve the tours found.
- The options are:
  - 2-opt
  - 2.5-opt
  - 3-opt

# ACOTSP

## Options: Local search

- 2-opt
  - Heuristic: Select two edges and exchange them (2-exchange)
  - Repeat this process for all the edges combinations looking for improvement




- 3-opt follows the same idea using 3 edges, also 2-opt moves are evaluated.
- 2.5-opt: Evaluates the insertion of a node coming from edge (A-B) between the nodes of other edge (C-D). Ex. A-C-B-D