CMU 15-781
Lecture 25:
Swarm Intelligence I

Teacher:
Gianni A. Di Caro
COLLECTIVE BEHAVIORS IN NATURE’S SYSTEMS
COLLECTIVE BEHAVIORS IN NATURE’S SYSTEMS
COLLECTIVE BEHAVIORS IN NATURE’S SYSTEMS

[Images of termite mounds and diagrams showing structural components like ventilation shaft, chimney, fungus combs, nursery galleries, and royal cell.]
WHAT ALL THESE BEHAVIORS HAVE IN COMMON?

- Distributed “society” of autonomous individuals/agents
- Control is fully distributed among the agents
- Communications among the individuals are localized
- Interaction rules and information processing seem to be simple: minimalist agent capabilities and interaction protocols
- System-level behaviors appear to transcend the behavioral repertoire of the single agent
- Deliberative and/or self-organizing cooperation is at work
- Local information propagates in a multi-step fashion

The overall response of the system features:
- Robustness
- Adaptivity
- Scalability

Swarm Intelligence design applies these same principles to obtain these same objectives as in Nature’s complex adaptive systems
A relatively novel research field (~25 years) that deals with **collective behaviors** resulting from the **local interactions** of (many) individual (minimalist) **units** with each **other** and with their **environment**

**Modeling:**
Study of collective behaviors in **natural and social systems**

**Engineering:**
*Bottom-up design* of distributed systems
Ontogenetic and phylogenetic evolution has (necessarily) followed a bottom-up approach (grassroots) to “design” systems:

- Instantiation of the basic units (atoms, cells, organs, organisms, individuals, . . . ) composing the system and let them (self-)organize to generate more complex/organized system-level behaviors and/or structures

- Population + Interaction protocols are “more important” than the single modules

- System-level structural patterns and behaviors are “emerging” properties

From an engineering point of view we can also choose a top-down approach:

- Acquisition of comprehensive knowledge about the problem/system to deal with, analysis, decomposition, definition of a possibly optimal strategy

- Amenable to formal analysis, “predictable” response
APPLICATIONS OF SI

• Combinatorial and global continuous optimization

• Distributed network control (routing)

• Clustering, data mining

• Reinforcement learning (policy learning)

• Multi/Swarm robotic systems
CHALLENGES OF SI DESIGN

- Characteristics/skills of the agents
- Size of the population (related to previous choice + “costs”)
- Neighborhood definition
- Interaction protocols and information to exchange
- Where the information is updated (agent, channel, environment)
- Use or not of randomness (or, heuristic decisions)
- Synchronous or asynchronous activities and interactions
- ...

Lots of parameters

Predictability and efficiency are important issues

Is a top-down approach better?
Yes when everything is stationary, “known”, and “tractable”

SI approaches are typically heuristics / meta-heuristics
Different ways of modeling communications, connection topology, and spatial distribution have given raise to different SI frameworks

- **Point-to-point communication (one-to-one):** two agents get in direct contact (e.g., antennation, trophallaxis, axons and dendrites in neurons)

- **Limited-range information broadcast (one-to-many):** the signal propagates to some limited extent throughout the environment and/or is available for a short time (e.g., fish’ use of lateral line to detect water waves, visual detection)

- **Indirect communication:** two individuals interact indirectly when one of them modifies the environment and the other responds to the modified environment, maybe at a later time (e.g., stigmergic, pheromone communication in ant colonies)

- **Physical mobility:** individuals move through the states of the environment, such as the connection topology changes over time (based on communication capability), and different areas of the environment are accessed in parallel by different agents

- **Static positioning, state evolution:** connection topology and/or positioning in the environment do not change over time. Local information propagates in multi-hop modality. The internal state of an individual changes over time.
SI ALGORITHMIC FRAMEWORKS (AND RELATIVES)

✦ Stigmergy, Mobility → **Ant Algorithms** and in particular to **Ant Colony Optimization (ACO)** [Dorigo & Di Caro, 1999], which is based on the shortest path finding abilities of ant colonies

✦ Stigmergy → **Cultural Algorithms** [Reynolds, 1994], population-based algorithms derived from processes of cultural evolution and exchange in societies

✦ Limited broadcast, Mobility → **Particle Swarm Optimization (PSO)** [Kennedy & Eberhart, 2001], related to fish schooling and bird flocking behaviors

✦ Point-to-point → **Hopfield neural networks** [Hopfield, 1982], derived from brain’s structure and behavior

✦ Point-to-point and neighbor limited broadcast → **Cellular Automata** [Wolfram, 1984], **Gossip algorithms** [Demers et al., 1987] derived from infection models

✦ Different combinations of communication and mobility → **Swarm robotics**, Adaptive network routing, Consensus algorithms

✦ **Genetic algorithms**, **Artificial immune systems**, . . .
ROAD MAP

• Ant Colony Optimization (ACO) metaheuristic
  • Stigmergy
  • ACO for Combinatorial optimization problems (TSP)
  • ACO for network problems

• Cellular Automata (maybe, a brief intro)

• Particle Swarm Optimization (PSO)

• Ant algorithms for clustering

• Swarm robotics fun
Stigmergy is at the core of most of all the amazing collective behaviors exhibited by the ant/termite colonies (nest building, division of labor, structure formation, cooperative transport).

P. Grassé (1959) introduced the term to explain nest building in termite societies (from the Greek stigma: sting and ergon: work, incite to work!): A stimulating configuration triggers a building action of a termite worker, transforming the configuration into another configuration that may trigger in turn another (possibly different) action by the same or other termites.

Stigmergy: any form of *indirect communication* among a set of (possibly) concurrent and distributed agents which happens through acts of *local modification of the environment* and *local sensing* of the outcomes of these modifications.

**Stigmergic variables:** The local environment’s variables whose value determine in turn the characteristics of agents’ response.

The presence of stigmergic variables is “expected” (depending on parameter setting) to give raise to *self-organized global behaviors or structural patterns* (e.g., nest building, chaining).

Stigmergic communication and control mechanisms in social insects have been reverse engineered to give raise to a multitude of *ant (colony) inspired algorithms*.

Best analogy: Blackboard/Post-it style of asynchronous communications.
Stigmergy leading to **diverging group behavior**: each agent has a different *threshold* to respond to the presence and the value of a stigmergic variable

- Distribution of labor
- Automatic task allocation
- Specialization of work

**Examples:**
- The height of a pile of dirty dishes floating in the sink (Everybody)
- Nest energy level in foraging robot activation (Krieger and Billeter, 1998)
- Level of customer demand in adaptive allocation of pick-up postmen, clustering of objects (Bonabeau et al., 1997, Lumer and Faieta, 1994)
Stigmergy leading to **converging group behavior**: the majority of the agents converge performing the same task or showing the same behavior.

- Stigmergic variable: **Intensity of pheromone trails** in ant foraging → Convergence of the colony on the shortest path between the nest and sources of food (Goss, Aron, Deneubourg, and Pasteels, 1989)

- While walking or touching objects, ants release a **volatile** chemical substance, called **pheromone**
- Pheromone distribution modifies the environment (the way it is perceived by other ants) creating a sort of attractive potential field for the ants
Use of ant colony inspire pheromone-based shortest path finding is at the core of the work of the Ant Colony Optimization metaheuristic.
PHEROMONE LAYING-FOLLOWING EXPERIMENTS

- Binary bridge with equal branches (Denebourg et al., 1990)

\[
P_U(m+1) = \frac{(U_m + r)^\alpha}{(U_m + r)^\alpha + (L_m + r)^h} \quad P_L(m+1) = 1 - P_U(m+1), \quad m = U_m + L_m
\]

- The number of ants that are on the upper and lower branch quantifies the amount of pheromone deposit on the branch → **Attraction towards the branch**
- \(r\) quantifies a the tendency towards a purely exploratory choice (volatility)
- \(\alpha\) biases the decision towards the branch with higher pheromone deposits
- \(r = 20, \alpha = 2\) fits real ants data
- With unequal branches, ants converge on the SP with a rate depending on \(\Delta\text{length}\)
SHORTEST PATHS WITH PHEROMONE LAYING-FOLLOWING

#Pheromone on a branch ∝ Frequency of fw/bw crossing ∝ Length (quality) of paths
• \( n \) decision states/nodes, \( x_1, x_2, \ldots, x_n \)

• A path (solution) is constructed as through a sequence decisions issued at each state according to a stochastic decision policy \( \pi_{\xi}(x_k; \tau^k, \eta^k) \)

• Pheromone \( \tau^k \) and heuristic \( \eta^k \) are real-valued local information parameter arrays

• Multiple ants iterating path construction

• \( \rightarrow \) Monte Carlo sampling: \( N \) joint probability distributions parametrized by \( \tau \) and \( \eta \) variable arrays
A (traveling) cost is associated with state transitions, costs are additive.

Once completed a solution:
- The sampled solution is evaluated (e.g., sum of the individual costs).
- "Credit" is assigned to each individual decision belonging to the solution.
- The value of the pheromone variables $\tau^k$ associated to each decision in the solution are modified according to the "credit."
- Pheromone values can also decay/change for other reasons (e.g., evaporation).
- Pheromone values locally encode how good is to take decision $i$ vs. $j$ as collectively estimated/learned by the agent/ant population through repeated solution sampling.

Pheromone distribution biases path construction.

Outcomes of path construction are used to modify pheromone distribution.

Form of Generalized Policy Iteration.
procedure ACO_metaheuristic()
    while (¬ stopping_criterion)
        schedule_activities
            ant_agents.construct_solutions.using.pheromone();
            pheromone_updating();
            daemon_actions(); /* optional */
        end schedule_activities
    end while
    return best_solution_generated;
ANT BEHAVIOR
SOLUTION CONSTRUCTION AND PHEROMONE UPDATING

\[ x_t = (c_i, c_j, ..., c_t) \]

\[ \Delta \tau(J(s_k)) \]

\[ \tau(c_i, c_{N-1}) \]

\[ \eta(c_i, c_{N-1}) \]
Given $G(V, E)$ find the Hamiltonian tour of minimal cost: NP-Hard

Every cyclic permutation of $n$ integers is a feasible solution:

$$\pi_1 = (1, 3, 4, 2, 6, 5, 7, 1), \quad \pi_2 = (2, 3, 4, 5, 6, 7, 1, 2)$$

$$c(\pi_2) = d_{23} + d_{34} + d_{45} + d_{56} + d_{67} + d_{71} + d_{12} = 93$$

Read also as set of edges:

$$\{(2,3), (3,4), (4,5), (6,7), (7,1), (1,2)\}$$

It’s easier to consider fully connected graphs, $|E| = |V|\cdot|V-1|$: If two nodes are not connected, $d$ is infinite.

“Related” combinatorial optimization problems: VRPs, SOP, TO, QAP, …
ACO FOR THE TRAVELING SALESMAN PROBLEM (TSP)

- **Pheromone variables**: $\tau_{ij} \in \mathbb{R}^+$ expresses how beneficial is (estimated, up to now) to have edge $(i,j)$ in the solution to optimize final tour length $\rightarrow |E|$ variables

- **Heuristic values** $\eta_{ij} \in \mathbb{R}^+$: problem costs $c_{ij} \in \mathbb{R}^+$ for traveling from $i$ to $j$ $\rightarrow |E|$ variables

Solution construction strategies (no repair, no look-ahed)

- **Extension**: when ant $k$ is in city $i$, how good is expected to include (feasible) city $j$ (next in the solution sequence $x^k(t)$)? $\rightarrow f(\tau_{ij}, \eta_{ij})$

- **Insertion**: how good is expected to insert (feasible) edge $(m,p)$ in the partial solution $x^k(t)$? $\rightarrow f(\tau_{mp}, \eta_{mp})$
Initialize $\tau_{ij}(0)$ to small random values and let $t = 0$;

**repeat**

Place $n_k$ ants on randomly chosen origin nodes;

**foreach** ant $k = 1, \ldots, n_k$ **do**

Construct a tour $x^k(t)$ [Update pheromone step-by-step];
Evaluate tour $x^k(t)$;

**end**

**foreach** [selected] edge $(i, j)$ of the graph **do**

Pheromone evaporation;

**end**

**foreach** [selected] ant $k = 1, \ldots, n_k$ **do**

**foreach** [selected] edge $(i, j)$ of $x^k(t)$ **do**

Update $\tau_{ij}$ using tour evaluation results;

**end**

**end**

Daemon actions [Local search];

$t = t + 1$;

**until** stopping condition is true;

**return** best solution generated;
Transition probability:

\[ p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t)\eta_{ij}^\beta(t)}{\sum_{u \in \mathcal{N}_i^k(t)} \tau_{iu}^\alpha(t)\eta_{iu}^\beta(t)} & \text{if } j \in \mathcal{N}_i^k(t) \\ 0 & \text{if } j \not\in \mathcal{N}_i^k(t) \end{cases} \]

where
\( \tau_{ij} \) represents the \textit{a posteriori} effectiveness of the move from node \( i \) to node \( j \)
\( \eta_{ij} \) represents the \textit{a priori} effectiveness of the move from \( i \) to \( j \) – desirability of the move

Other “most common” transition rule in ACO implementations:

\[ p_{ij}^k(t) = \frac{\alpha \tau_{ij}(t) + (1 - \alpha)\eta_{ij}(t)}{\sum_{u \in \mathcal{N}_i^k(t)} \left( \alpha \tau_{iu}(t) + (1 - \alpha)\eta_{iu}(t) \right)} \]
A balance between pheromone intensity, $\tau_{ij}$, and heuristic information, $\eta_{ij}$

- If $\alpha = 0$:
  - No pheromone information is used, i.e. previous search experience is neglected
  - The search then degrades to a stochastic greedy search

- If $\beta = 0$:
  - The attractiveness of moves is neglected
  - The search algorithm is similar to SACO

Heuristic information adds an explicit bias towards the most attractive solutions, e.g.

\[ \eta_{ij} = \frac{1}{d_{ij}} \]
To improve exploration abilities, and to prevent premature convergence:

$$\tau_{ij}(t) \leftarrow (1 - \rho)\tau_{ij}(t)$$

with $\rho \in [0, 1]$

- $\rho$ specifies the rate at which pheromones evaporate, causing ants to "forget" previous decisions
- $\rho$ controls the influence of search history
- For large values of $\rho$, pheromone evaporates rapidly, while small values of $\rho$ result in slower evaporation rates
- Large values therefore implies more exploration, more random search
Pheromone is iteratively deposited in an additive cumulative modality based on solution quality.

$$\tau_{ij}(t + 1) = \tau_{ij}(t) + \sum_{k=1}^{n_k} \Delta \tau_{ij}^k(t)$$

where

$$\Delta \tau_{ij}^k(t) = \frac{1}{L^k(t)}$$

$L^k(t)$ is the length of the path constructed by ant $k$ at time step $t$.

$n_k$ is the number of ants.
QUESTIONS

1. Why an additive, cumulative rule for pheromone updating and not an average, for instance? (not looking for averages, but for the “sparse” best solutions)

2. Is there any potential problem with pheromone bounds? (get to zero, unlimited growth)

3. Is there any potential problem of premature convergence?

4. Is it a good idea to have a large number of samples / ants given the adopted rule for pheromone updating? (all solutions do pheromone updating → A lot of “bad” ones!)

5. How do we balance policy evaluation and policy improvement?
Idea: assign credits relative to some $Q$ constant value related to problem’s costs

$Q$ = an upper bound estimate on the length of the optimal tour, in Ant-cycle

$Q$ = small value related to the range of cost values, Ant-density & Ant-Quantity

- Three variations in the way pheromone deposits are calculated
- Ant-cycle AS:

$$\Delta \tau_{ij}^k(t) = \begin{cases} \frac{Q}{f(x^k(t))} & \text{if link } (i,j) \text{ occurs in path } x^k(t) \\ 0 & \text{otherwise} \end{cases}$$

- Ant-density AS:

$$\Delta \tau_{ij}^k(t) = \begin{cases} Q & \text{if link } (i,j) \text{ occurs in path } x^k(t) \\ 0 & \text{otherwise} \end{cases}$$

- Ant-quantity AS:

$$\Delta \tau_{ij}^k(t) = \begin{cases} \frac{Q}{d_{ij}} & \text{if link } (i,j) \text{ occurs in path } x^k(t) \\ 0 & \text{otherwise} \end{cases}$$
The best ants add pheromone proportional to quality of their paths

\[ \tau_{ij}(t + 1) = \tau_{ij}(t) + \Delta \tau_{ij}(t) + n_e \Delta \tau_{ij}^e(t) \]

where

\[ \Delta \tau_{ij}^e(t) = \begin{cases} \frac{Q}{f(\tilde{x}(t))} & \text{if } (i, j) \in \tilde{x}(t) \\ 0 & \text{otherwise} \end{cases} \]

e is the number of elite ants
\( \tilde{x}(t) \) is the current best route

Objective is to direct the search of all ants to construct a solution to contain links of the current best route(s)
• ACS addresses main AS’ shortcomings and introduces new components
• A different transition rule is used
• A different pheromone update rule is defined
• Step-by-step local pheromone updates are introduced
• Candidate lists are used to favor specific nodes and save a lot of computation (at each step, check among
• $n \ll |E|$ possible decisions, $|E|$ can easily be $10^N$, $N > 3$)
• Later (and more performing) versions make use of a daemon component based on local search
ACS: TRANSITION RULE

- The pseudo-random-proportional action rule:

\[
  j = \begin{cases} 
  \arg \max_{u \in \mathcal{N}_i^k(t)} \{ \tau_{iu}(t) \eta_{iu}^\beta(t) \} & \text{if } r \leq r_0 \\
  J & \text{if } r > r_0 
  \end{cases}
\]

where \( r \sim U(0, 1) \), and \( r_0 \in [0, 1] \) is a user-specified parameter.

- \( J \in \mathcal{N}_i^k(t) \) is a node randomly selected according to probability

\[
p_{ij}^k(t) = \frac{\tau_{ij}(t) \eta_{ij}^\beta(t)}{\sum_{u \in \mathcal{N}_i^k} \tau_{iu}(t) \eta_{iu}^\beta(t)}
\]

\( \mathcal{N}_i^k(t) \) is a set of valid nodes to visit.
Transition rule creates a bias towards nodes connected by short links and with a large amount of pheromone.

Parameter $r_0$ is used to balance exploration and exploitation:
- if $r \leq r_0$, the algorithm exploits by favoring the best edge.
- if $r > r_0$, the algorithm explores.
- the smaller the value of $r_0$, the less best links are exploited, while exploration is emphasized more.

The transition rule is the same as that of AS when $r > r_0$. 
We are looking for the best, not the “average”

- Global update rule:
  - Only the globally best ant, $x^+(t)$, is allowed to reinforce pheromone concentrations on the links of the corresponding best path

  $$\tau_{ij}(t+1) = (1 - \rho_1)\tau_{ij}(t) + \rho_1\Delta\tau_{ij}(t)$$

  where

  $$\Delta\tau_{ij}(t) = \begin{cases} 
  \frac{1}{f(x^+(t))} & \text{if } (i,j) \in x^+(t) \\
  0 & \text{otherwise}
  \end{cases}$$

  with $f(x^+(t)) = |x^+(t)|$, in the case of finding shortest paths

  - Favors exploitation
  - $x^+(t)$ as the iteration-best vs global-best
• **Persistence, conservative approach:** For small values of $\rho_1$, the existing pheromone concentrations on the edges evaporate slowly, while the influence of the best route is dampened.

• **Volatile, aggressive approach:** For large values of $\rho_1$, previous pheromone deposits evaporate rapidly, but the influence of the best path is emphasized.

• The effect of large $\rho_1$ is that previous experience is neglected in favor of more recent experiences $\rightarrow$ more exploration.

• **Simulated Anneling approach:** If $\rho_1$ is adjusted dynamically from large to small values, exploration is favored in the initial iterations of the search, while focusing on exploiting the best found paths in the later iterations.
A “good” choice is potentially made locally “less good” after being selected. This is to favor exploring other local choices during the same iteration loop.

- **Local update rule:**
  - Applied by each ant as soon as a new link is added to the path:
    \[ \tau_{ij}(t) = (1 - \rho_2)\tau_{ij}(t) + \rho_2\tau_0 \]
  - with \( \rho_2 \) also in (0, 1), and \( \tau_0 \) is a small positive constant

Pheromones don’t go to zero!
\( \mathcal{N}_i^k(t) \) is organized to contain a list of candidate nodes

Candidate nodes are preferred nodes, to be visited first

Let \( n_l < |\mathcal{N}_i^k(t)| \) denote the number of nodes in the candidate list

The \( n_l \) nodes closest to node \( i \), i.t.o. cost, are included in the candidate list and ordered by increasing distance

When a next node is selected, the best node in the candidate list is selected

If the candidate list is empty, then node \( j \) is selected from the remainder of \( \mathcal{N}_i^k(t) \)
## ACS: (OLD) PERFORMANCE (1997)

<table>
<thead>
<tr>
<th>Problem name</th>
<th>ACS</th>
<th>GA</th>
<th>EP</th>
<th>SA</th>
<th>Optimum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eil50</td>
<td>425 (427.96)</td>
<td>428 (N/A)</td>
<td>426 (427.86)</td>
<td>443 (N/A)</td>
<td>425 (N/A)</td>
</tr>
<tr>
<td></td>
<td>(1,830)</td>
<td>[25,000]</td>
<td>[100,000]</td>
<td>[68,512]</td>
<td></td>
</tr>
<tr>
<td>Eil75</td>
<td>535 (542.37)</td>
<td>545 (N/A)</td>
<td>542 (549.18)</td>
<td>580 (N/A)</td>
<td>535 (N/A)</td>
</tr>
<tr>
<td></td>
<td>(3,480)</td>
<td>[80,000]</td>
<td>[325,000]</td>
<td>[173,250]</td>
<td></td>
</tr>
<tr>
<td>KroA100</td>
<td>21,282 (21,285.44)</td>
<td>21,761 (N/A)</td>
<td>N/A</td>
<td>N/A</td>
<td>21,282 (N/A)</td>
</tr>
<tr>
<td></td>
<td>(4,820)</td>
<td>[103,000]</td>
<td>[N/A]</td>
<td>[N/A]</td>
<td></td>
</tr>
</tbody>
</table>

### Additional Table

<table>
<thead>
<tr>
<th>Problem name</th>
<th>ACS best integer length (1)</th>
<th>ACS number of tours generated to best</th>
<th>ACS average integer length</th>
<th>Standard deviation</th>
<th>Optimum (2)</th>
<th>Relative error</th>
<th>CPU sec to generate a tour</th>
</tr>
</thead>
<tbody>
<tr>
<td>d198</td>
<td>15,888</td>
<td>585,000</td>
<td>16,054</td>
<td>71</td>
<td>15,780</td>
<td>0.68 %</td>
<td>0.02</td>
</tr>
<tr>
<td>(198-city problem)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pcb442</td>
<td>51,268</td>
<td>595,000</td>
<td>51,690</td>
<td>188</td>
<td>50,779</td>
<td>0.96 %</td>
<td>0.05</td>
</tr>
<tr>
<td>(442-city problem)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>att532</td>
<td>28,147</td>
<td>830,658</td>
<td>28,523</td>
<td>275</td>
<td>27,686</td>
<td>1.67 %</td>
<td>0.07</td>
</tr>
<tr>
<td>(532-city problem)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rat783</td>
<td>9,015</td>
<td>991,276</td>
<td>9,066</td>
<td>28</td>
<td>8,806</td>
<td>2.37 %</td>
<td>0.13</td>
</tr>
<tr>
<td>(783-city problem)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fl1577</td>
<td>22,977</td>
<td>942,000</td>
<td>23,163</td>
<td>116</td>
<td>[22,204 - 22,249]</td>
<td>3.27 + 3.48 %</td>
<td>0.48</td>
</tr>
<tr>
<td>(1577-city problem)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
• At the end of each iteration, a local search is applied to all tours built by the ants
• The resulting iteration (or global so far) best tour gets pheromone updating
• Selected LS: 3-Opt
• Computationally expensive, but rewarding!
Two edges, \((i,j)\) and \((l,k)\), are selected, removed, and replaced by two other edges \((i,k)\) and \((j,l)\) (or, \((k,i)\), \((l,j)\))

One of the two paths needs to get reverted!

Gain: \((i,k) + (j,l) - (i,j) - (k,l)\)

\(n(n-1) = O(n^2)\) possible successors in the 2-exchange neighborhood

→ quadratic search complexity for each single 2-opt step move
• Including the initial solution, as well as 2-opt moves, there is a total of $2^3$ feasible rewirings for each selected triple of edges
• $n(n - 1)(n - 2) = O(n^3)$ successors
• One move does not revert the path $\rightarrow$ appropriate for asymmetric TSP
Adapts AS to include a local search using tabu search

Global update rule is changed such that each ant’s pheromone deposit on each link of its constructed path is proportional to the quality of the path:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \left(\frac{\rho}{f(x^k(t))}\right) \left(\frac{f(x^-(t)) - f(x^k(t))}{f(\hat{x}(t))}\right)$$

- $f(x^-(t))$ is the cost of the worst path found so far
- $f(\hat{x}(t))$ is the cost of the best path found so far
- $f(x^k(t))$ is the cost of the path found by ant $k$
Global update is similar to that of ACS
- If based on only the global-best path, may exploit too much
- If based on only the iteration-best, more exploration
- Used mixed strategies
- At point of stagnation, all $\tau_{ij}$ are initialized to max value, after which iteration-best is applied for a number of iterations.

Point of stagnation:

$$\sum_{i \in V} \frac{\lambda_i}{n_G} < \epsilon, \quad \epsilon > 0$$

where $\lambda_i$ is the number of links leaving node $i$ with $\tau_{ij}$-values greater than $\lambda \delta_i + \tau_{i,min}$. $\delta_i = \tau_{i,max} - \tau_{i,min}$

$$\tau_{i,min} = \min_{j \in \mathcal{N}_i} \{\tau_{ij}\}$$

$$\tau_{i,max} = \max_{j \in \mathcal{N}_i} \{\tau_{ij}\}$$
Clamping of pheromone:

- If after application of the global update rule $\tau_{ij}(t + 1) > \tau_{\text{max}}$, $\tau_{ij}(t + 1)$ is explicitly set equal to $\tau_{\text{max}}$
- If $\tau_{ij}(t + 1) < \tau_{\text{min}}$, $\tau_{ij}(t + 1)$ is set to $\tau_{\text{min}}$
- Upper bound helps to avoid stagnation. How?
- What is the advantage of having a lower pheromone limit?

Local update, applied by each ant after adding a new link to the path:

$$\tau_{ij}(t + 1) = \tau_{ij}(t) + \Delta \tau_{ij}(t)$$
Stagnation still occurred, due to large differences between min and max pheromones.

Smoothing strategy used to reduce the differences between high and low pheromone concentrations.

At point of stagnation, all pheromone concentrations are increased proportional to the difference with the maximum bound:

\[ \Delta \tau_{ij}(t) \propto (\tau_{\text{max}}(t) - \tau_{ij}(t)) \]

Stronger pheromone concentrations are proportionally less reinforced than weaker concentrations.

Increases the chance of links with low pheromone intensity to be selected as part of a path, and thereby increases the exploration abilities of the algorithm.
ACO SUMMARY

• Reverse engineering of stigmergic pheromone laying-following mechanisms in ant colonies
• Monte Carlo sampling (MCMC), Generalized policy learning
• A number of different heuristic recipes (common in SI and other heuristic optimization domains)
• State of the art performance (when coupled with LS)
• Guaranteed performance: yes, in the probabilistic limit
• Applied to a large variety of CO problems
• Hundreds of publications
• Applied in the real world: Barilla, Migros, port management, logistics, ....