Distributed Agent-Based Ant Colony Optimization for Solving Traveling Salesman Problem on a Partitioned Map

Sorin Ilie, Amelia Badica, ***Costin Badica***
University of Craiova, Romania

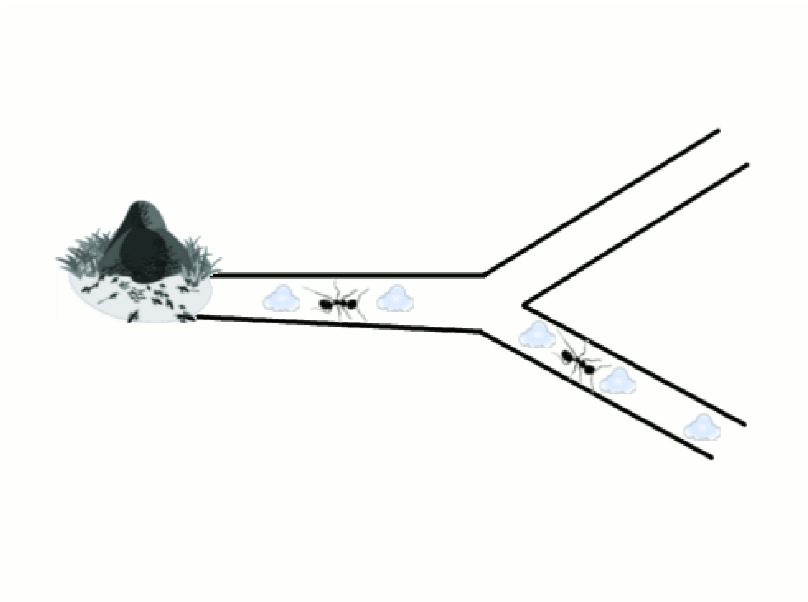
Overview

- Ant Colony Optimization
- Mathematical Model
- Distributed Architecture => ACODA
- The Traveling Salesman Problem
- Experimental results

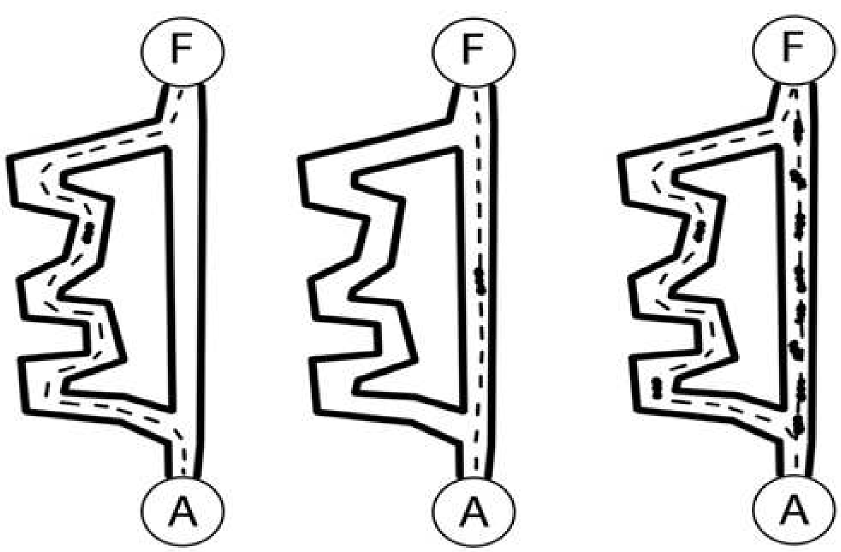
ACO – Random Search

ACO – Pheromone Deposit

ACO – Pheromone Guided Search



ACO – Convergence to Shortest Path



Approaches for distributing ACO

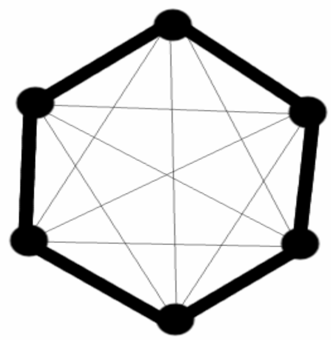
	Sequential	Fine-grained	Multi-colony	Agent-based
Environment	data	data	data	a set of agents
Ants	functions	functions	functions	a set of agents (sometimes mobile)
Messages	none	ants synchronize after each iteration	colonies synchronize with a given frequency	ants communicate with each other and with nodes
Ant Movement	one at a time	all at once synchronized	asynchronous (but one at a time in each colony)	asynchronous
Observations		large overhead	currently best approach	large overhead

Motivation

- Use of multi-agent systems for modeling ants' environment.
  - It was observed that complexity of ants' movement stems from the complexity of the environment.
- Mapping of ants' environment to a distributed architecture and the mapping of the ants' migration to messages exchanged between the agents located in the ants' environment.
  - n agents, N ant migrations/any 2 agents, cost of ant message on a single machine = a and between 2 machines = $b>a$.
  - Execution time on 1 machine $T_1 = a\, N\, n\, (n\text{-}1)/2$ and on n machines $T_n = b\, N\, (n\text{-}1)$. If $n>2b/a$ then $T_1 > T_n$.

Traveling Salesman Problem

- Given a weighted graph, the goal is to find the shortest tour that visits each node exactly once.



Probabilistic Choices

$$p_{i,j} = \frac{(\tau_{i,j}^{\alpha})(\eta_{i,j}^{\beta})}{\sum(\tau_{i,j}^{\alpha})(\eta_{i,j}^{\beta})}$$

where:

- $\tau_{i,j}$ = *amount of pheromone deposited* on edge (i,j)
- α = parameter to control the influence of $\tau_{i,j}$
- $\eta_{i,j}$ = *desirability* of edge (i,j) computed as the inverse of the weight $w_{i,j}$ of edge (i,j), i.e.
- β = parameter to control the influence of $\eta_{i,j}$
- j = a node reachable from node i that was not visited yet

Pheromone Increment

$$\Delta\tau_{i,j}^{k} = \begin{cases} 1/L_k & \text{if ant k travels on edge(i, j)} \\ 0 & \text{otherwise} \end{cases}$$

where:

- L_k is the cost of the k-th ant tour.
- $\Delta\tau$ is the amount of pheromone ant k deposits on edge (i,j)

Pheromone Deposit

$$\tau_{\mathrm{i,j}} = (1 - \rho)\tau_{\mathrm{i,j}} + \rho\Delta\tau_{i,j}^{k}$$

where:

- $\tau_{i,j}$ is the amount of pheromone on edge (i,j)
- $\Delta\tau$ is the amount of pheromone ant k deposits on edge (i, j)
- ρ is the evaporation rate $0 \leq \rho < 1$

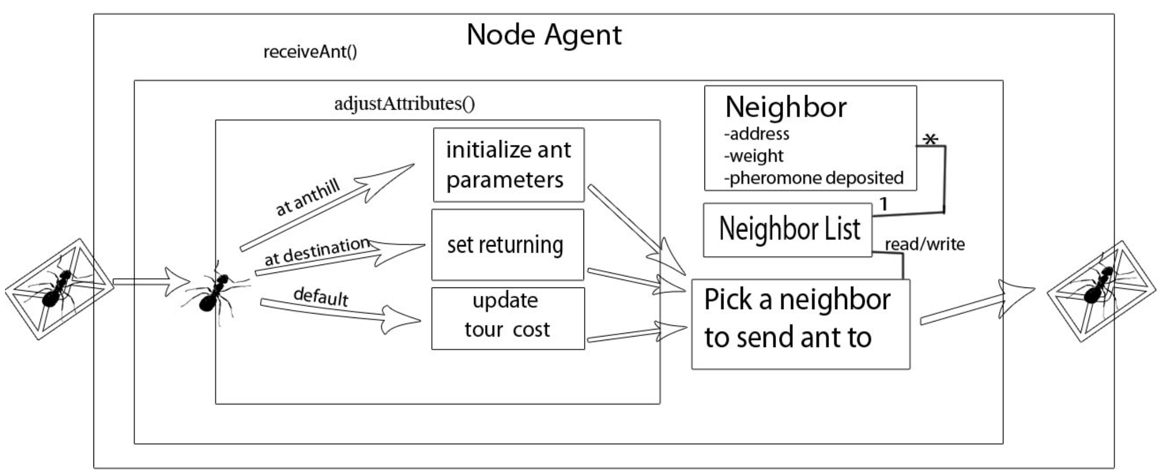
Local Evaporation

$$\tau_{i,j} = (1 - \zeta)\,\tau_{i,j} + \zeta\,\tau_0$$

where:

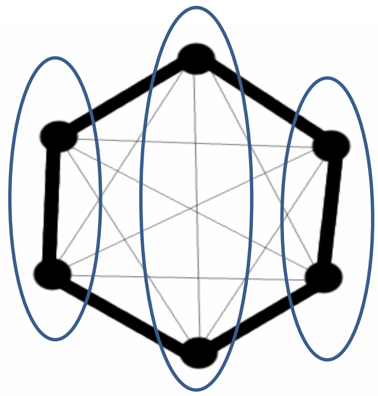
- ζ is the evaporation rate $0 \leq \zeta < 1$
- τ_0 is the initial amount of pheromone on each edge

Architecture



Traveling Salesman Problem

- In our approach each agent manages a set of nodes



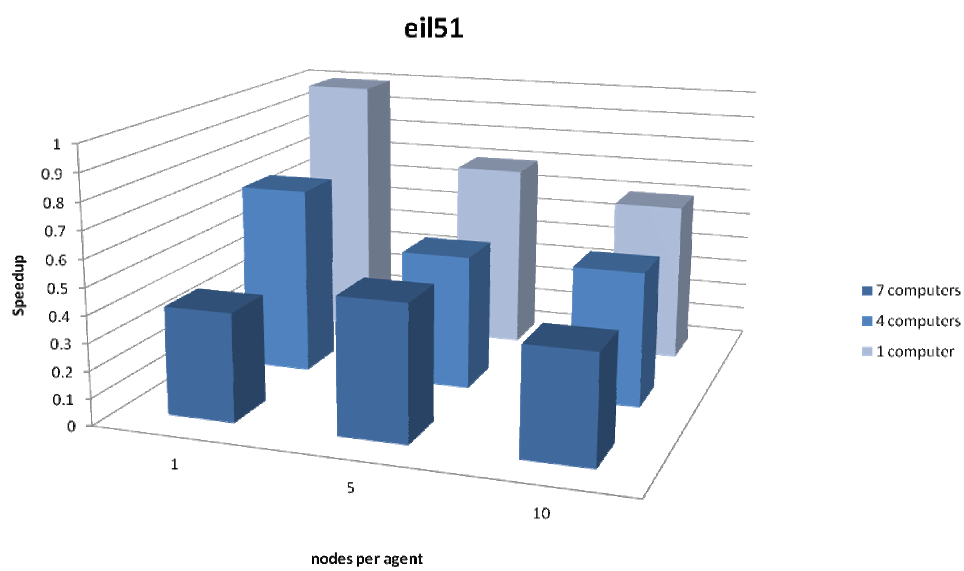
Experimental Setup

- Experiment = a fixed number of independent rounds
- Round = one execution of ACODA for given set of input parameters
- Initialization
  - Creation of JADE containers, 1 container/machine
  - Partition the map, i.e. allocate nodes to agents
  - Create agents and distribute them to containers
- Execution
  - Stops when a given number M of ant moves are recorded by a reference node

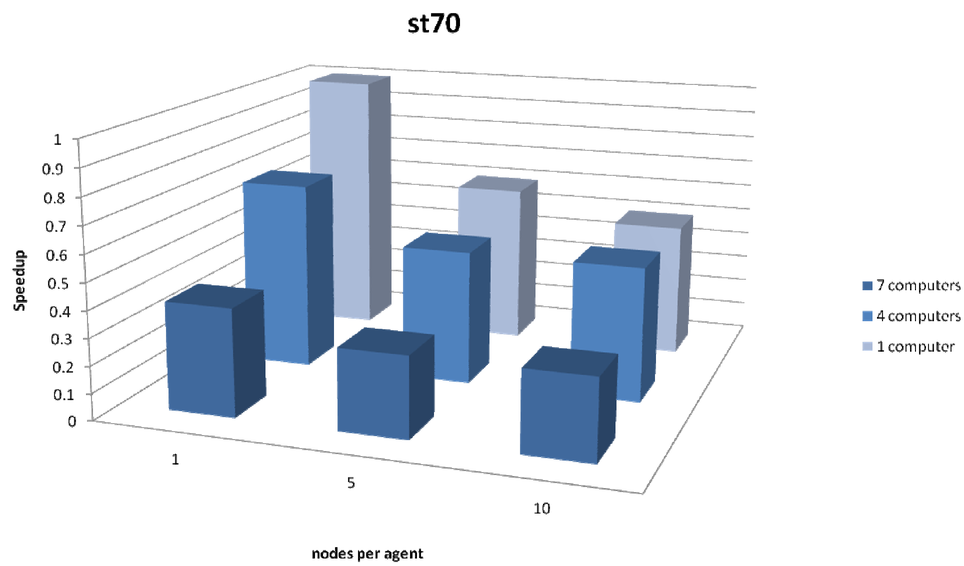
Parameters and Network

- Benchmarks from TSPLIB
  - eil51, st70, kroA100, ch150, gr666
  - $n \in \{51, 70, 100, 150, 666\}$
- $\tau_0 = 1/(n^2 w_{avg})$
- $\rho = \zeta = 0.1, \alpha = 1, \beta = 5$
- $M = 10000$
- Network:
  - 1, 4, and 7 computers with dual core processors at 2.5 GHz and 1GB of RAM memory
  - high-speed Myrinet interconnection network at 2Gb/s
  - variable number of nodes managed by each agent: $k \in \{1, 5, 10\}$

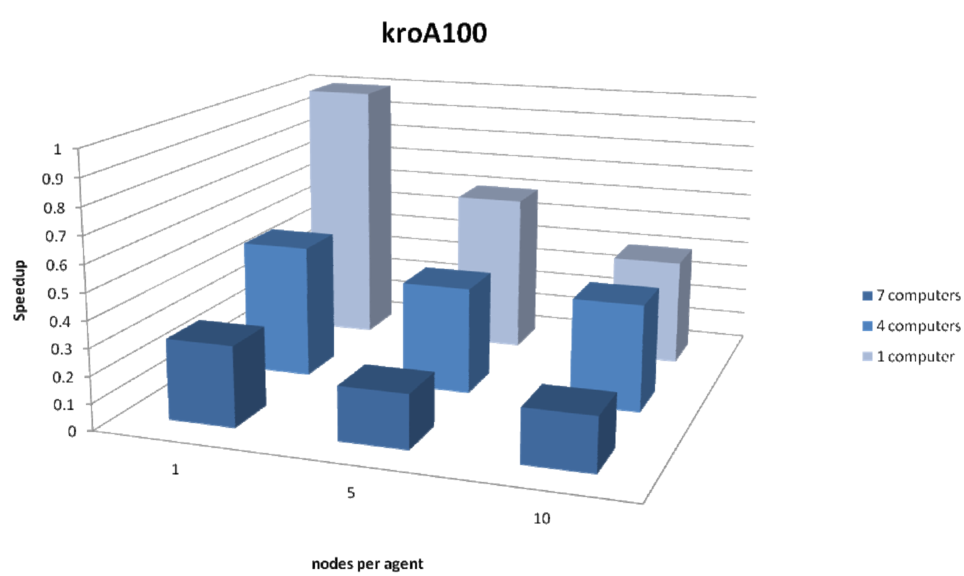
Experimental Results



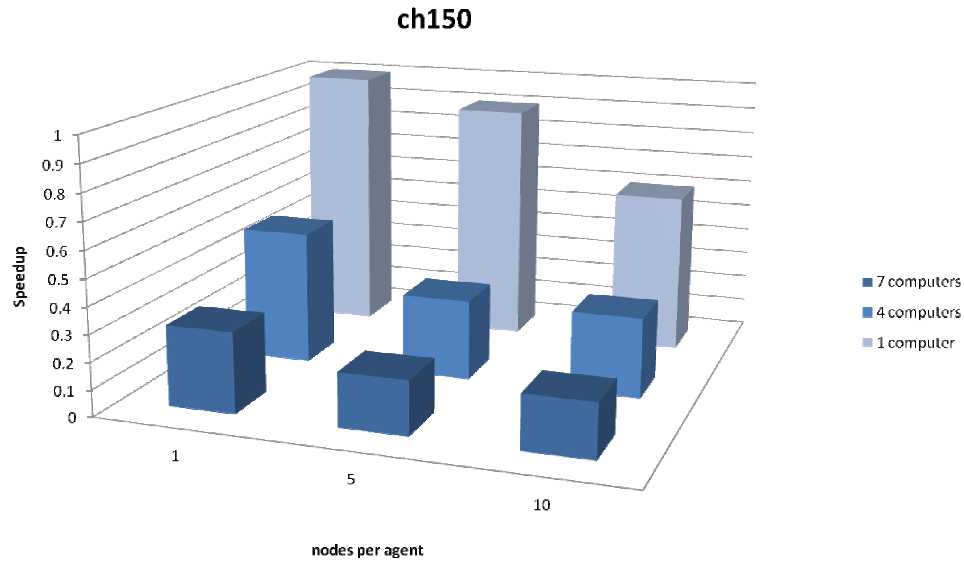
Experimental Results



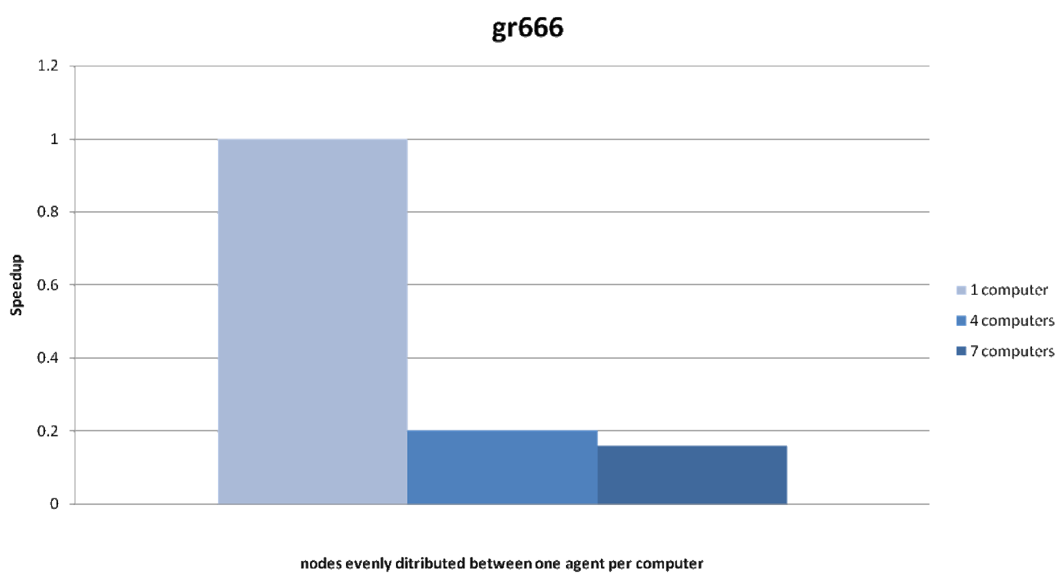
Experimental Results



Experimental Results



Experimental Results

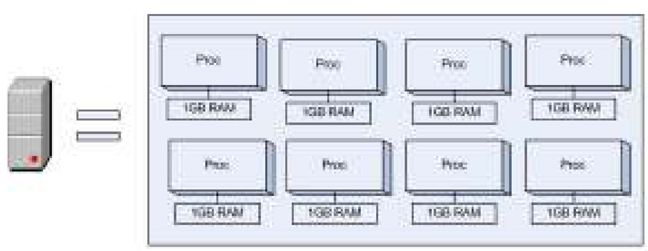


Recent experiments

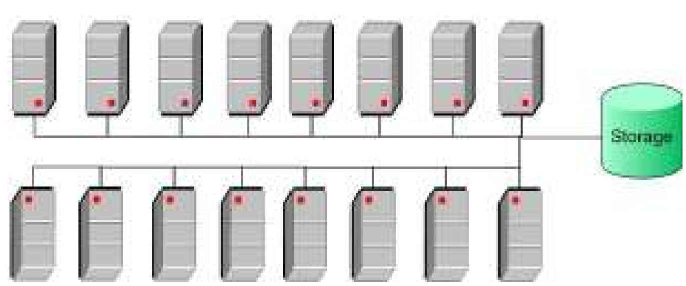
- Experiments were also ran on an **Infragrid cluster consisting of 128 cores**, each one with 1GB of RAM, connected by an Infiniband 40 Gb/s network.

The Cluster

- Each server has 8 cores and a single IP. Each core can be used as an individual machine with 1GB RAM.

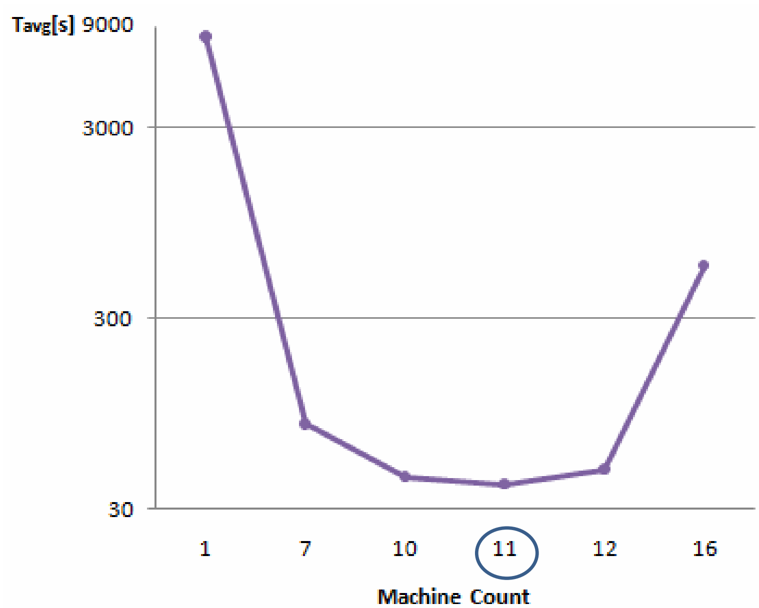


- There are 16 servers connected to a single central storage space



Experiments on Infragrid cluster

gr666



Conclusions

- Experiments with ACODA for partitioned TSP map on different networks.
- Future Works:
  - Increase the problem size.
  - Consider other versions of ACO.
  - Analyze the effect of the partitioning scheme.

Questions?