# Comparing Multi-agent System and Mixed-Integer Programming Approaches on a Simple Optimization Problem

P. Laroque F. Pirard F. Archambault M. Quoy P. Gaussier

ETIS, CNRS UMR8051 / ENSEA / Université de Cergy-Pontoise
F95000 France

ICAI '10 - 07/15/2010

# The Problem

- Numerous problems fall into the "search of an extremum" (minimum or maximum) of a given cost – or objective / profit – function problem
- Several approaches, including MIP, can solve the problem in the linear case
  - hard to predict computation time given the input parameters
  - sometimes a "good" solution is enough if one can get it from a faster algorithm
  - unsufficient robustness when faced with great variations in the model dynamics
- MAS approach, based on self-organization principles, may be an answer to such limitations
  - however solution is not optimal. Is it "good"?

# The Problem

- Numerous problems fall into the "search of an extremum" (minimum or maximum) of a given cost – or objective / profit – function problem
- Several approaches, including MIP, can solve the problem in the linear case
  - hard to predict computation time given the input parameters
  - sometimes a "good" solution is enough if one can get it from a faster algorithm
  - unsufficient robustness when faced with great variations in the model dynamics
- MAS approach, based on self-organization principles, may be an answer to such limitations
  - however solution is not optimal. Is it "good"?

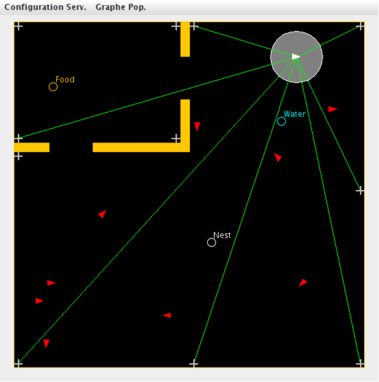
# Our Test Case

- Warehouse location problem: distribute a set of products to a set of client areas from production plants and / or warehouses
- production plant capacity is sufficient to satisfy the customers' requests; warehouse capacity is limited
- We want to know:
  - which warehouse is used and where it is located
  - the product flows (plant/warehouse, plant/customer, warehouse/customer)

# MIP Model

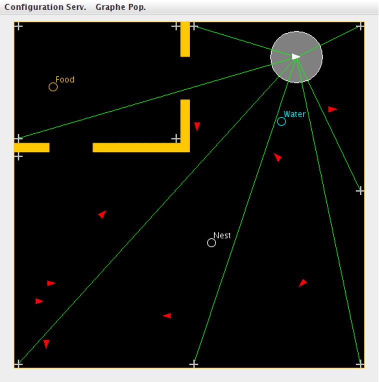
- Objective function: sum of delivery and warehouse costs (to minimize)
- Decision variables:
  - existence of a warehouse at a given location (binary)
  - flows (plant - warehouse, plant - customer, warehouse - customer)
- Constraints:
  - satisfy customer area requests
  - respect warehouse capacity
  - respect conservation of flows
- Solving algorithm: Branch-and-bound

# MAS Model



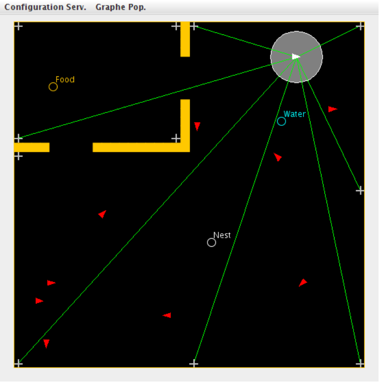

- motivation-driven agents
- vision "sensor", landmarks & place cells
- "comfort" threshold ; falling below minimum comfort value triggers planning strategy
- "cognitive" agents embed a cognitive map to retrieve complex paths leading to resources
- hebbian learning rules on cognitive map weights allow for learning new paths and forgetting obsolete ones

# MAS Model



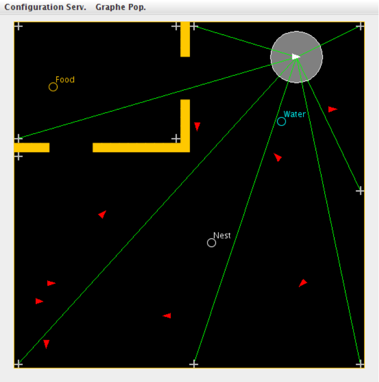

- motivation-driven agents
- vision "sensor", landmarks & place cells
- "comfort" threshold ; falling below minimum comfort value triggers planning strategy
- "cognitive" agents embed a cognitive map to retrieve complex paths leading to resources
- hebbian learning rules on cognitive map weights allow for learning new paths and forgetting obsolete ones

# MAS Model



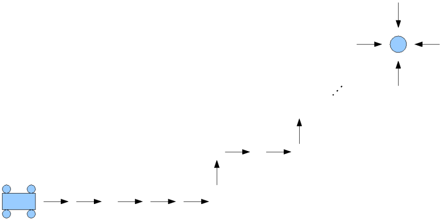

- motivation-driven agents
- vision "sensor", landmarks & place cells
- "comfort" threshold ; falling below minimum comfort value triggers planning strategy
- "cognitive" agents embed a cognitive map to retrieve complex paths leading to resources
- hebbian learning rules on cognitive map weights allow for learning new paths and forgetting obsolete ones

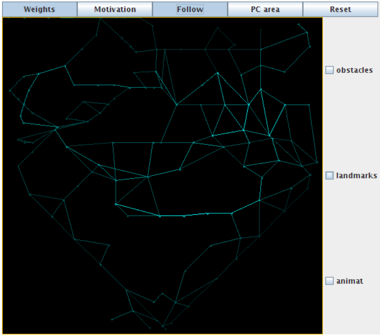
# MAS Model




- motivation-driven agents
- vision "sensor", landmarks & place cells
- "comfort" threshold ; falling below minimum comfort value triggers planning strategy
- "cognitive" agents embed a cognitive map to retrieve complex paths leading to resources
- hebbian learning rules on cognitive map weights allow for learning new paths and forgetting obsolete ones

# Agent Navigation





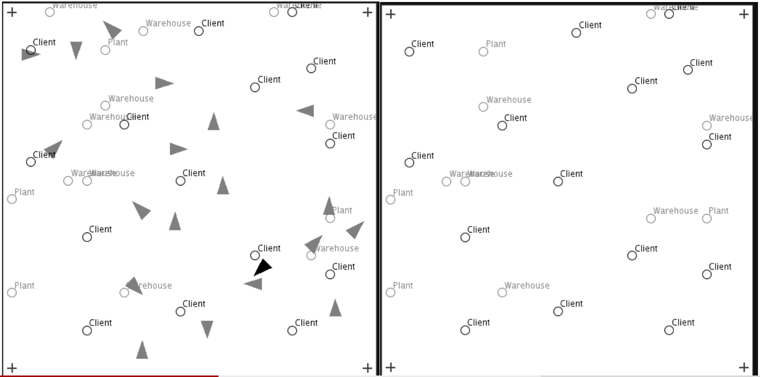
# Warehouse Creation

- Agents can transport small quantities of goods, and store them when / where they want
- Transport and deposit decisions are probabilistic, based on observations of certain social insects behavior: $P(n)$ (prob. for deposit when $n$ units of resources) is

$$P(n) = 1 - (1 - p)^n$$

# Experiments

- Fixed resources (plants)
- Environment = 20x20 square grid, cell = (empty, plant, warehouse, customer)
- 10 to 20 agents ("trucks") per test run
- results averaged over ten successive runs (due to randomness in agents behavior)



# Computing the Results

1. minimum distance from $W_i$ to a MAS "correspondent" $W'_i$:

$$d_i = \min_j \left( d \left( W_i, W'_j \right) \right)$$

2. average distance for a given experiment $Exp_k$:

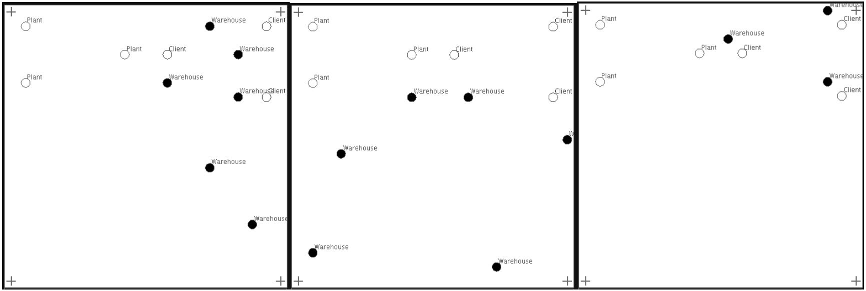
$$d(Exp_k) = \left( \sum_{i=1}^{n} \frac{d_i}{n} \right) + p \left( \left| n' - n \right| \right)$$

3. average results on $N_e$ experiments:

$$d(MAS, MIP) = \frac{1}{N_e} \sum_{i=1}^{Ne} d(Exp_i)$$

# First Results

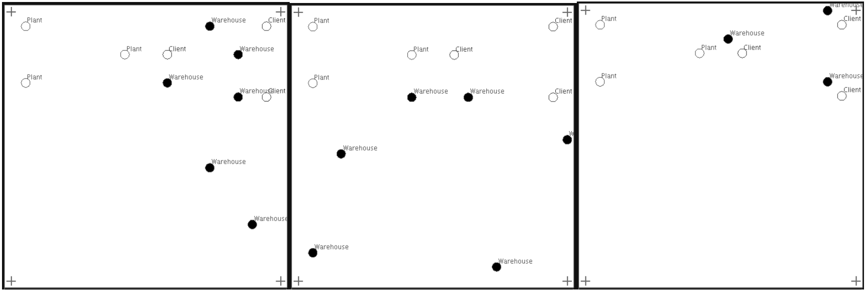
- The MAS solution (left) is compared to both the MIP solution (best possible, right) and a randomly generated solution (middle):




| | MAS solution | random solution |
|---|---|---|
| average distance | 19.6831 | 49.4644 |
| standard deviation | 8.24 | 15.73 |

# First Results

- The MAS solution (left) is compared to both the MIP solution (best possible, right) and a randomly generated solution (middle):




| | MAS solution | random solution |
|---|---|---|
| average distance | 19.6831 | 49.4644 |
| standard deviation | 8.24 | 15.73 |

# Perspectives

Next important issues:

1. complexity of the MAS approach when the environment size changes?
2. what happens to the MIP and MAS algorithms when things change with time?
    - MAS: cognitive map helps for agents adaptation
    - MIP: modeling technique under construction

# Summary

- Accuracy of a MAS approach to simple optimization problems
- MAS solution obtained after a short - and predictible - delay; "quality" not always predictible
- MIP approach gives best possible results, time needed not always predictible