(towards) A middleware to support stigmergy for cognitive agents

Luca Gardelli, Alessandro Ricci, Mirko Viroli,
Andrea Omicini, Enrico Oliva

DEIS – Cesena
Department of Electronics, Computer Science & Systems
Alma Mater Studiorum, Università di Bologna
{aricci, mviroli, aomicini, lgardelli, eoliva}@deis.unibo.it

Abstract

- Stigmergy principles have often been employed as effective means to engineer indirect coordination in MASs where agents were very simple.
- We would like to extend this approach to systems populated with cognitive agents, i.e. agents that can reason on a symbolic stigma.
- In this presentation we will give some insights on the on-going research about artefacts to support cognitive stigmergy. In particular we will provide some examples of a middleware based on TuCSoN technology.

Stigmergy

- Grassé coined the word *stigmergy* (1959) to name the process of coordination in termites society building their nests
- Currently we consider stigmergy as
  - A mechanism of (indirect) coordination through local modifications to a shared environment
- The pheromone plays the role of *stigma*: its "evolution" over space and time is regulated by evaporation, diffusion and aggregation processes

Agents in stigmergy

- When building system based on stigmergic coordination, agents are *very simple* – at least for the coordination task – e.g. sense pheromone field, put pheromone...
- ...in a certain sense pheromone encapsulate implicitly a goal...
- ...agents can have totally different goals but still be willing to cooperate...
- ..so what if the stigma becomes a piece of knowledge and agents are able to do more complex reasoning?

Cognitive agents

- In social sciences an agent which has mental processes, perception, reasoning is said to be *cognitive*
- We want to study the implications of extending stigmergic coordination to societies of cognitive agents..
- .. how to engineer stigmata to achieve better efficiency
- The main ingredients are
  1. An environment
  2. A symbolic stigma
  3. Stigma "evolution" rules

Mechanisms & Design Issues

1. The environment must provide a mechanism to store symbolic stigmata collected by local interaction with agents
2. The symbolic stigma must be interpretable by agents, although they may not share the same ontology
3. How stigma evolves over space and time?
   Where are these rules located?
   Does stigmergy rules still can be applied in this case?

Scenario: wikipedia

- Wikipedia is a free web encyclopedia
- The knowledge stored in Wikipedia is the result of a indirect coordination/interaction process which is stigmergic
- But agents are cognitive so we would loose something if analyzing that MAS adopting traditional stigmergy point of view
- Why not applying cognitive stigmergy principles to it?

Scenario: wikipedia

- The simplest mechanism to achieve stigmergy is to let user put (explicitly or not) annotations into the shared space – Wikipedia.
- Other users can see that annotations and add their owns affecting each other reasoning processes
- Annotations can be handled by the environment accordingly to cognitive stigmergy processes

Scenario: wikipedia

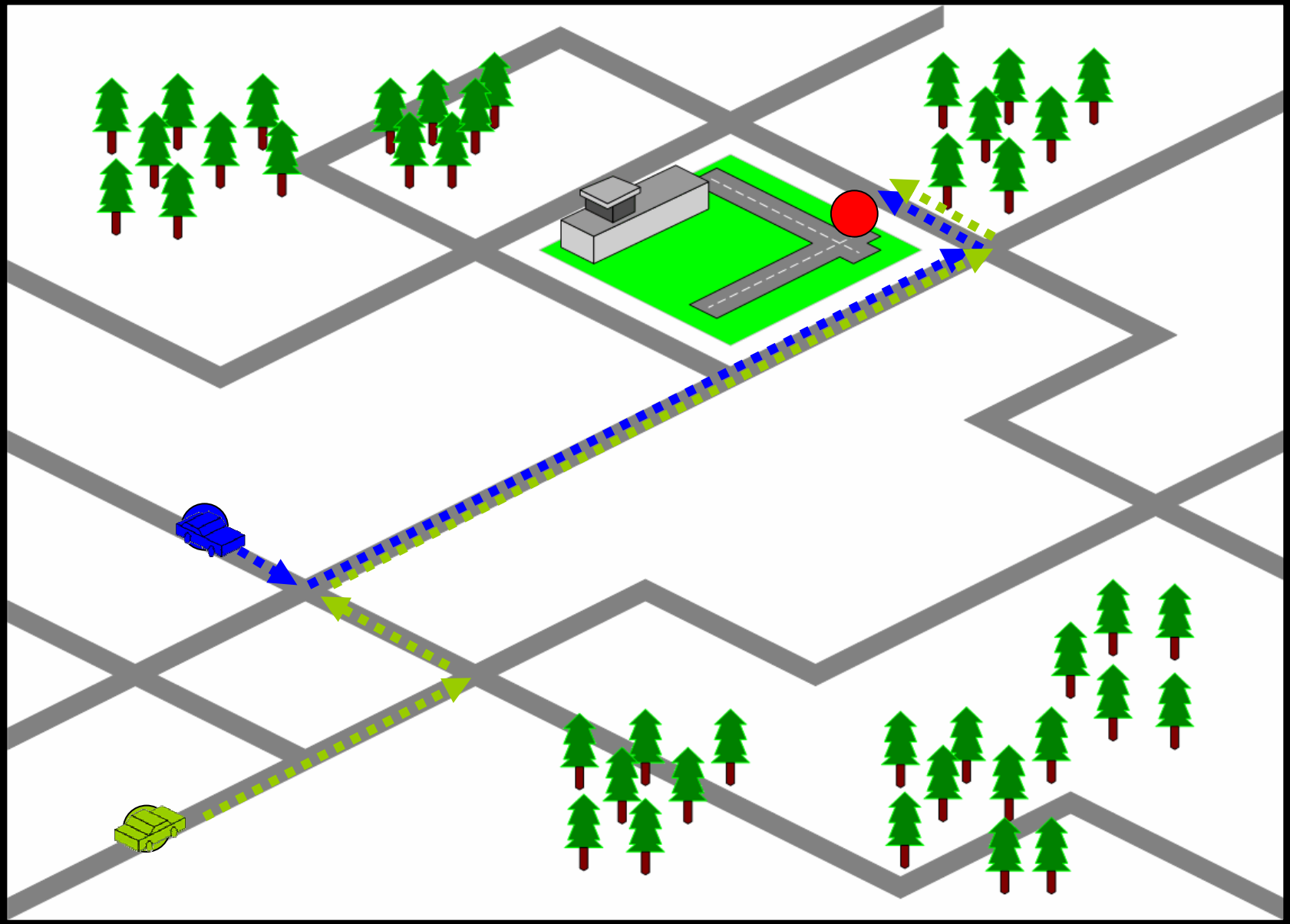
- This annotation can be used to represent several kinds of stigma..
  - ***N*** agents find ***X*** to be very useful – explicit
  - Agents that had look at ***X*** also had a look at *Y* implicit (e.g. Amazon online bookshop)
  - Agent *A* say "If you like *X* don't miss *Y*" - explicit
- Also the annotation can be effectively exploited for correction/revision purpose providing a more general abstraction

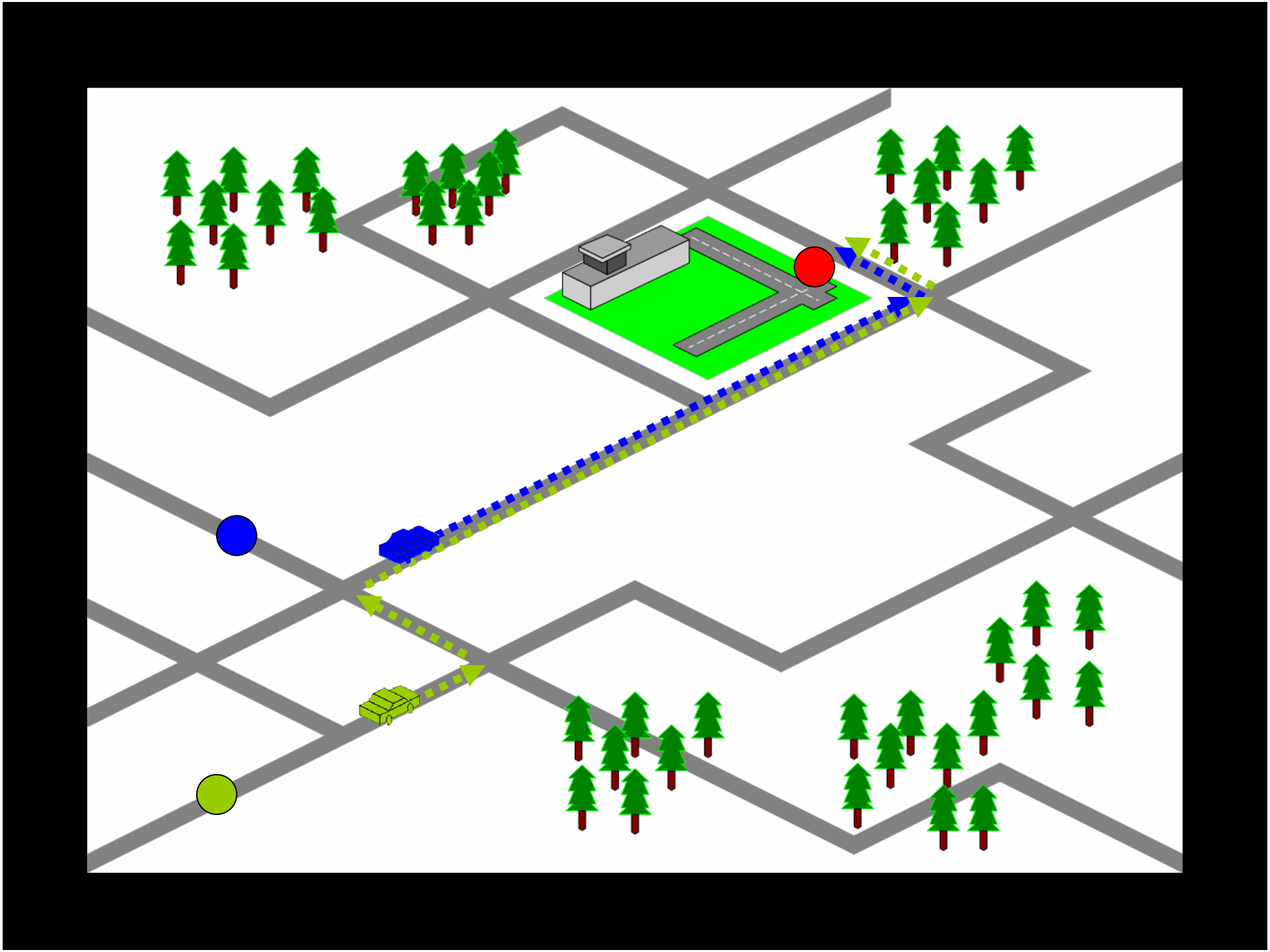
Applications

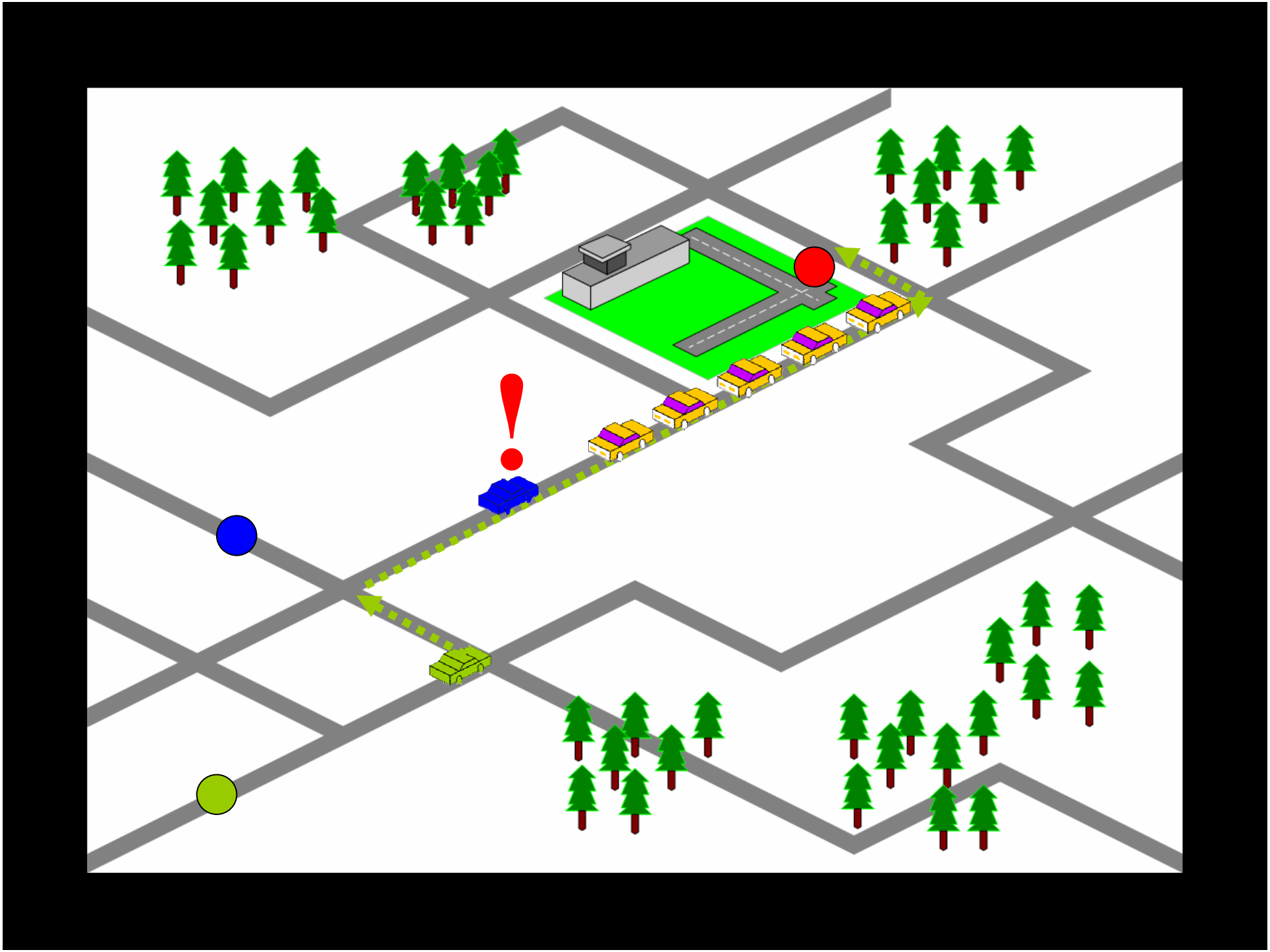
- Among the possible applications that could benefit of using annotation mechanism
  - e-learning
    - "if you can't solve this ex. you may find *Y* useful"
  - e-commerce
    - "users that bought ***X*** also have bought *Y*" – implicit
    - "I have bought X and it's doesn't work"- explicit
      (this kind already exists)
  - ambient intelligence
    - "I have used this coffee machine it has eaten my money!"
    - "This restaurant is fine and it is quite cheap!"
      I would have appreciated this info. here in Budapest ☺
  - traffic control
    - "Road X is congested all the time!"

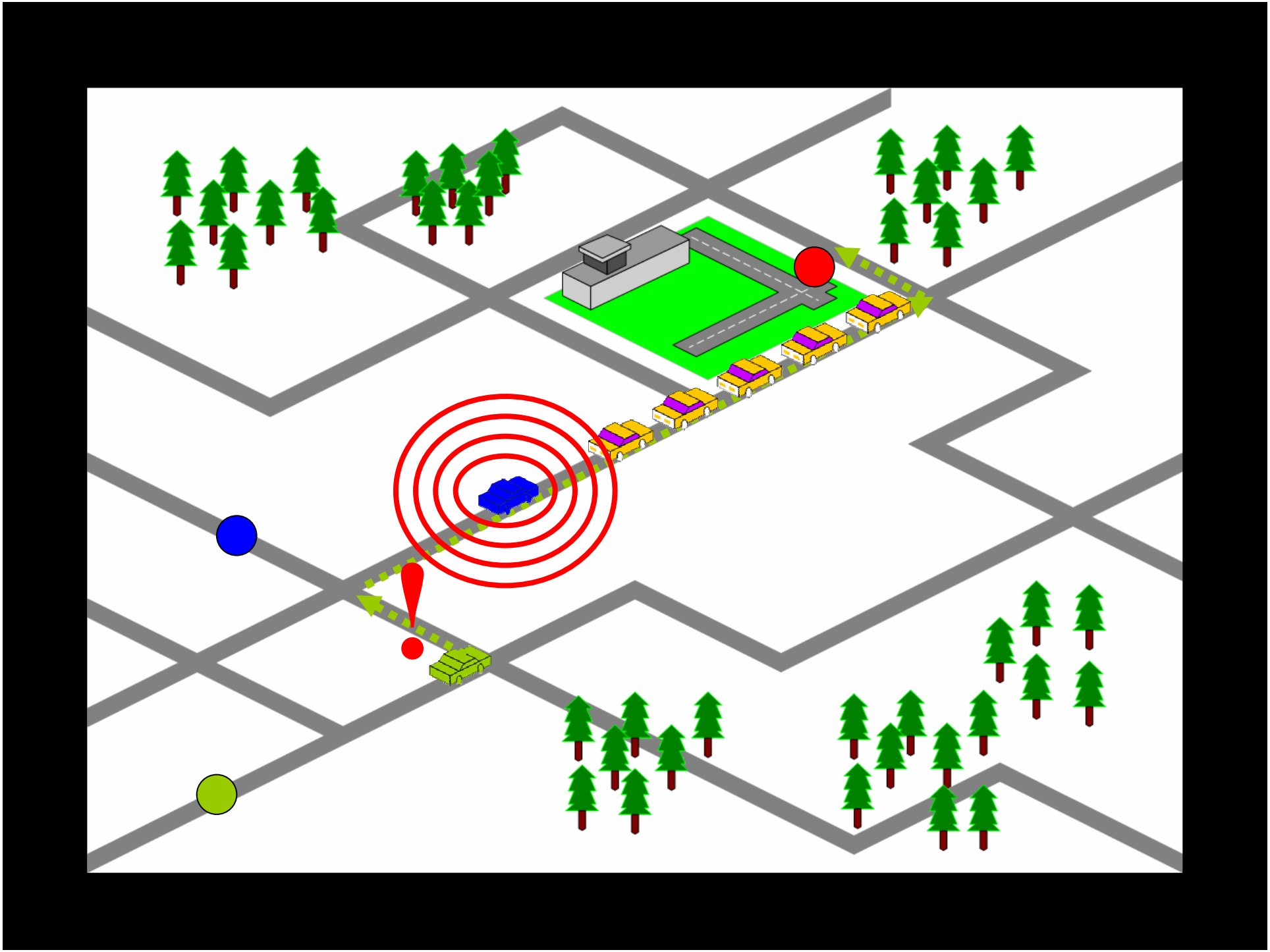
Scenario: Adaptive routing

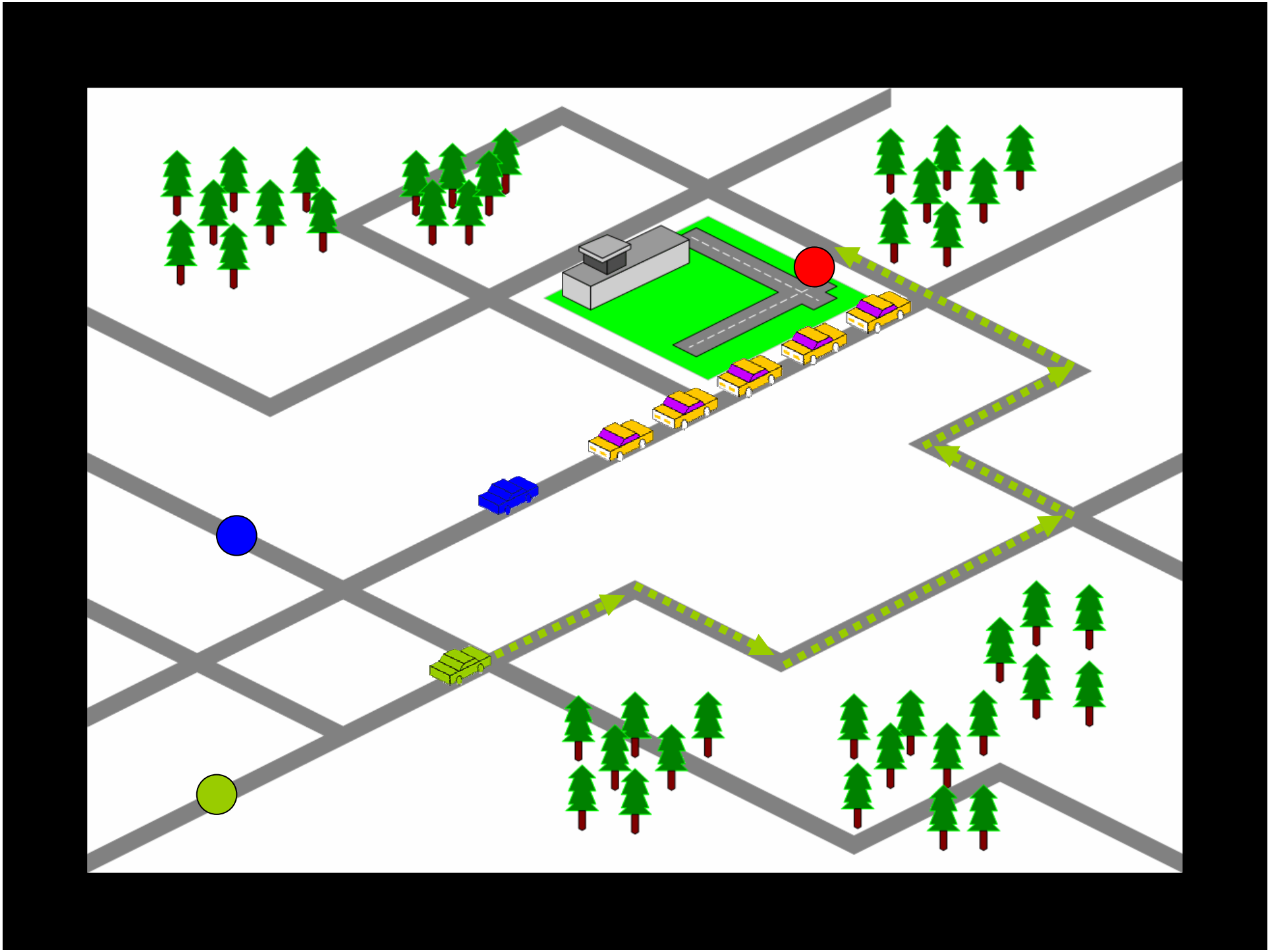
- Consider an application of adaptive routing for a business of couriers (package delivery)
- Although they may have a detailed roadmap, to increase the overall performance, couriers can share real-time information about traffic flow
- Couriers perceive that info and, in real-time, adapt their path accordingly to their strategy

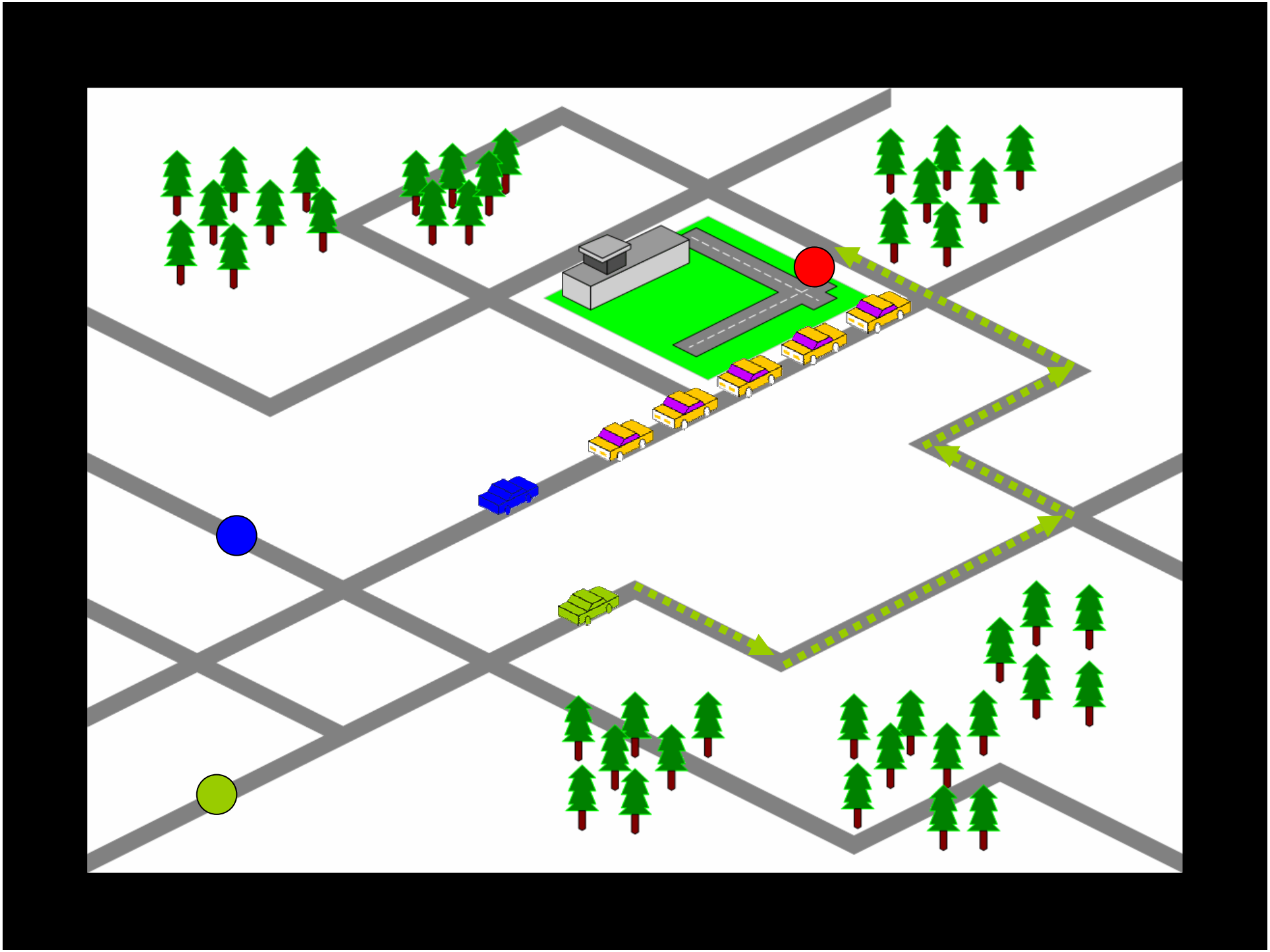










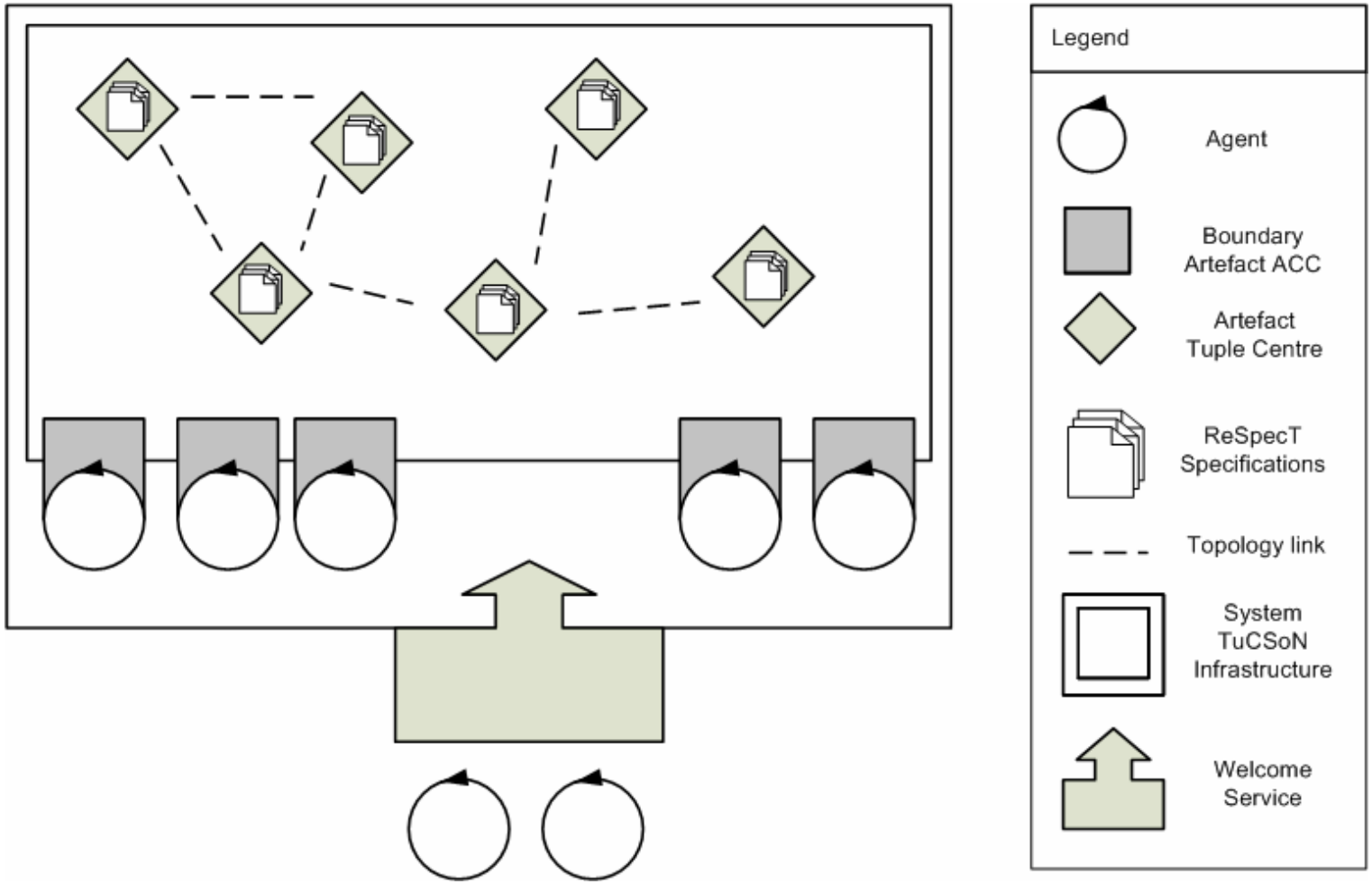


Our vision about MASs

- We promote the idea of MAS as a collection of agents and artefacts
- Artefacts – grounded on Activity Theory – are tools which may be exploited by agents for their own goals
- TuCSoN is our source of coordination artefacts
  - A tuple centre in which tuples are predicates of first order logic
  - Its behavior is programmable by ReSpecT language (for more details look for Omicini@PROMAS and Viroli@ENV)

Note: TuCSoN has been exploited for the meeting scheduler application ☺

MAS Architecture



Stigma and evolution rules

- Stigma is represented by 1° order logic tuples
- Evolution rules are encoded using ReSpecT language: the specifications might be changed dynamically
- When dealing with information there are some dimensions that are always interesting
  - Freshness
  - Reliability
  - Relevance..
- These dimensions might be quantifiable and change over time in a similar way pheromone does

Tuples

- In order to build this application we have to design either agents and environment behavior
- First we must decide which is the tuple template, e.g.

```
        road(n1,n2,c) – persistent
roadrt(n1, n2, c, f, r, re) – realtime
```

  where n1/n2 is the first/second node, c is the cost, f is the freshness, r is reliability and re is relevance
- Then we decide how to evolve the stigma over space and time...

Rules: freshness

- We assigned to the info a counter which decreases it over time (evaporation)
- If the same info is deposited the counter is reset to the max value (aggregation)
- If the counter reach zero the info is deleted (evaporation)

An example of stigma evolution

- Let's consider the tuple `road(n1, n2, c, f, r, re)`
- The reactions are trigger by a timer event and uses delta(L) parameter

```
reaction(out_r(timer), (
    in_r(road(N1,N2,C,F,R,RE)),
    rd_r(delta(L)),
    F1 is F - L,
    F1 > 0,
    out_r(road(N1,N2,C,F1,R,RE))
)).
reaction(out_r(timer), (
    in_r(road(N1,N2,C,F,R,RE)),
    rd_r(delta(L)),
    F1 is F - L,
    F1 =< 0
)).
```

Rules: reliability

- Premise: agents cannot lie!
- Reliability depends on how many agents share the “same” info
- You can count how many agents deposited the same info (aggregation)

Rules: relevance

- An information is relevant to you if it affects your decisions
- For example you can decide that a traffic information is no longer useful if it is X far away from its source
- So you would not spread this information from X on (diffusion)

Engineering efforts

- Using a Java library for Operation Research problems about graphs...
- Using TuCSoN infrastructure...
- Design phase - 2 hours
- The configurator deploys and configures tuple centres given the graph topology: written in Java exploiting TuCSoN APIs - 1 hour
- Evolution rules: written using ReSpecT language – 2 hours
  - More debugging tools are needed
  - Formalization in Stochastic π-Calculus in progress...

Conclusions

- We are currently studying the implications of extending the definition of stigmergy to cognitive agents, to do so...
- ..we have built a middleware prototype on top of TuCSoN technology
- We have programmed artefacts to act as an active environment which evolves stigmata
- Monitoring tools W.I.P.
- More details and discussions in *(if accepted)*

  Cognitive Stigmergy: A Framework Based on Agents and Artifacts, by Ricci, Omicini, Viroli, Gardelli, Oliva – MA4CS@ECCS'05

Future works

- We want to compare performances with a multi-pheromone infrastructure
  - But it's not all about performance it's also about architecture and methodologies!!
- We want to explore more scenarios: knowledge sharing, e-learning platforms, e-commerce..
- So we could better generalize/formalize our results..