KATHOLIEKE UNIVERSITEIT LEUVEN

DistriNet RESEARCH GROUP

Coordination and delegate MAS in large-scale distributed control applications

Prof. Tom Holvoet
DistriNet labs
Dept. of Computer Science
KULeuven, Belgium

Overview

1. Delegate MAS: BDI through the environment

   Tom Holvoet, Paul Valckenaers
   LNCS - Environments for MAS 2006
   Alexander Helleboogh, Danny Weyns, Tom Holvoet, Rutger Claes
   ITS Conference 2007
   IEEE Journal on ITS (submitted)

2. "Using Equation-Free Macroscopic Analysis for Studying Self-Organising Emergent Solutions"

   Giovanni Samaey, Tom Holvoet, Tom De Wolf
   IEEE Conf. on Self-Adaptation and Self-Organisation (SASO'2008)
   Venice, Italy

Multi-agent systems

- What is an agent?
  - A computer system situated in some environment that is capable of autonomous actions in the environment in order to achieve its design objectives[1]
- What is a MAS?
  - A set of interacting agents
  - Example application areas
    - E-business
    - Logistics, manufacturing control
    - Intelligent traffic systems

[1] **Wooldridge M. Agent-based software engineering. Software Engineering 144, 1997**

What are MAS?

- MAS are in essence
  - a solution strategy
  - a basis for a software architecture
  - for distributed problem solving
  - embedded in an environment
  - that is inaccessible, non-deterministic and dynamic

"systems designed to cope with dynamics"

→ **"Agents are *99% computer science*, and 1% AI."**

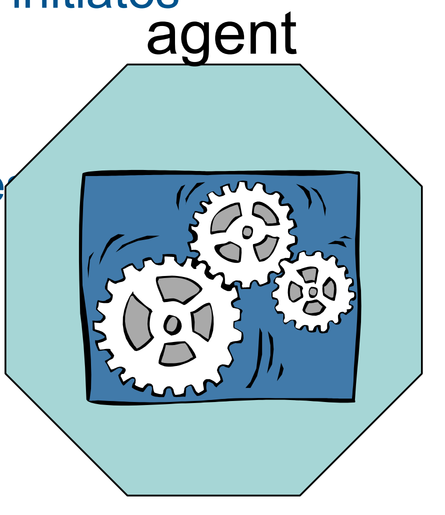
- NOT FOR FREE !

  "owning a hammer does not make one an architect"

Agents

→ **agents** need to integrate different behaviour aspects:

- **reactive**
  reacts to stimuli (changes in env., communication, …)
- **autonomous**
  does not require user interaction
- **proactive**
  aims to achieve its own goals, therefore initiates appropriate actions
- **social**
  cooperates / coordinates / communicates
- **embodied**
  situated in the environment
- **mobile**
  moves around network sites



Agent architectures

1. Deductive reasoning agents
   - 1956 – present
   - "Agents make decisions about what to do via ***symbol manipulation***. Its purest expression, proposes that agents use ***explicit logical reasoning*** in order to decide what to do."
2. Reactive / behaviour-based agents / situated MAS
   - 1985 – present
   - "Problems with symbolic reasoning lead to a reaction against this — lead to the ***reactive agents*** movement."
3. Practical reasoning agents
   - 1990 – present
   - "Agent use **practical reasoning** (towards actions, not towards beliefs) – beliefs / desires / intentions."
4. Hybrid agents
   - 1989 – present
   - *"Hybrid* architectures attempt to **combine** the best of reasoning and reactive architectures."

Practical Reasoning Agents

- *BDI - a theory of practical reasoning* – [Bratman, 1988]

→for "resource-bounded agent"

→Core concepts

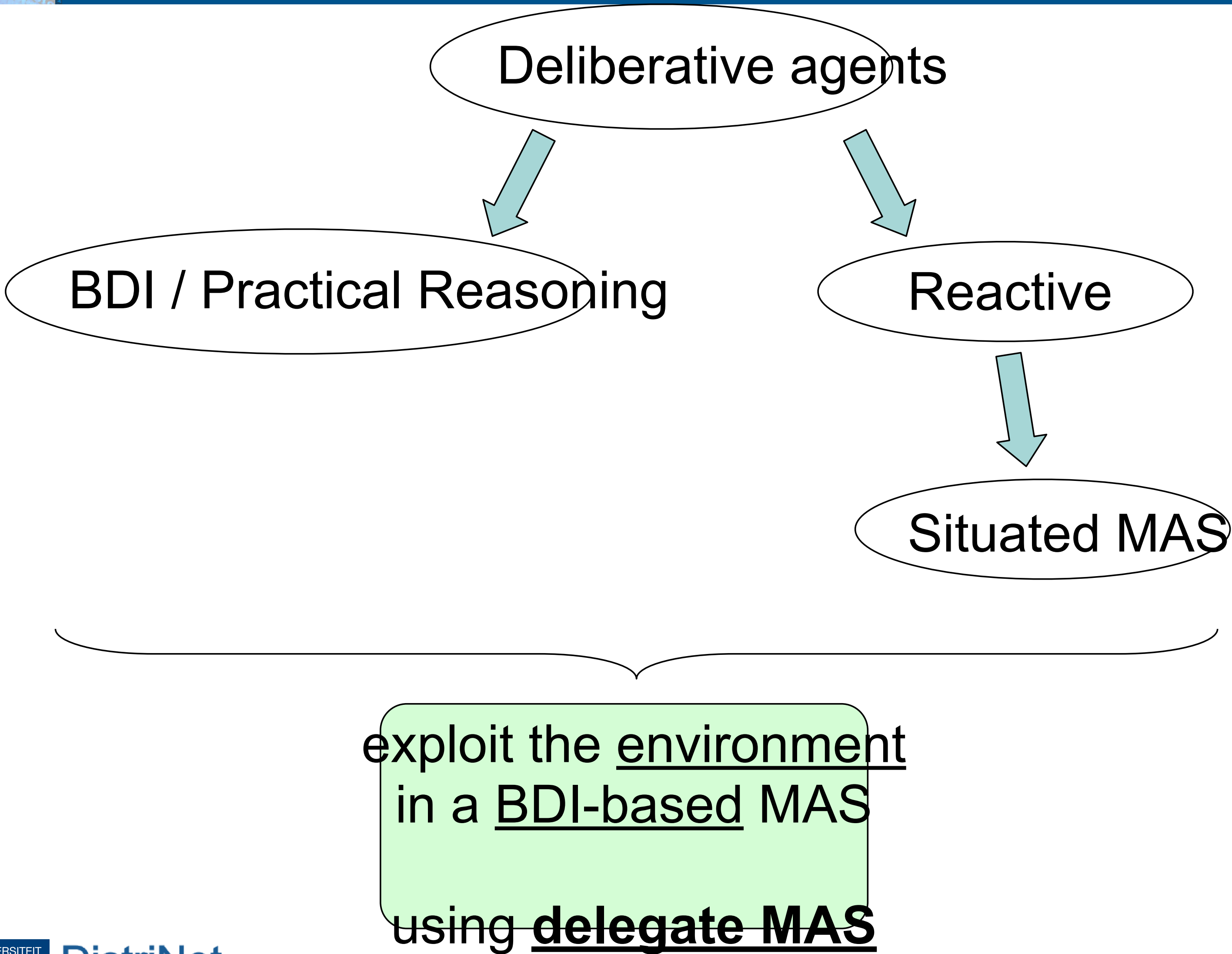
- Beliefs = information the agent has about the world
- Desires = state of affairs that the agent would wish to bring about
- Intentions = desires (or actions) that the agent has committed to achieve

Practical Reasoning Agents (cont.)

- agent control loop

```
while true
            observe the world;
            update internal world model;
            deliberate about what intention to achieve next;
            use means-ends reasoning to get a plan for the intention;
            execute the plan
end while
```

- when to reconsider intentions !?



KATHOLIEKE UNIVERSITEIT LEUVEN DistriNet RESEARCH GROUP

KATHOLIEKE UNIVERSITEIT LEUVEN

DistriNet RESEARCH GROUP

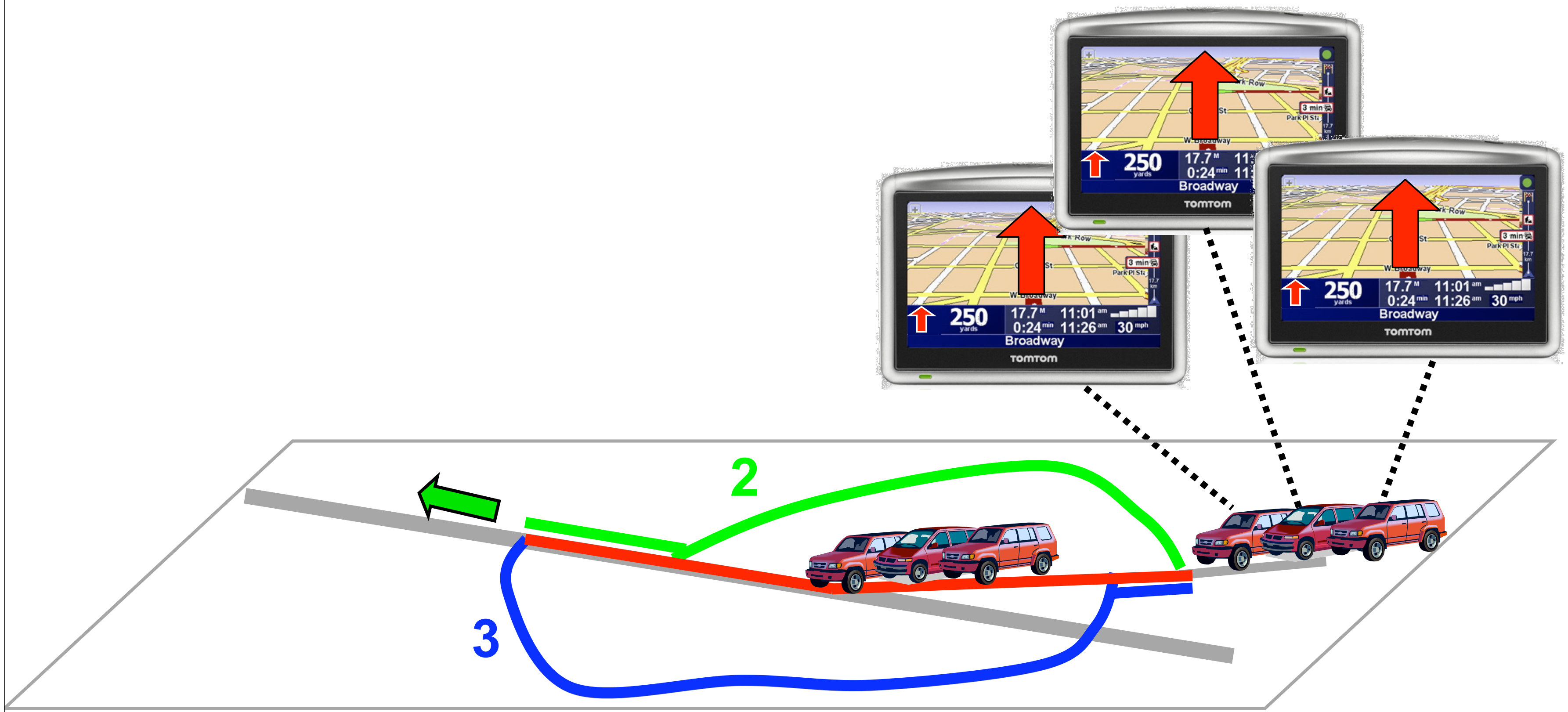
Delegate MAS:

BDI through the environment

Traffic Routing

Individual navigation

=> Congestion



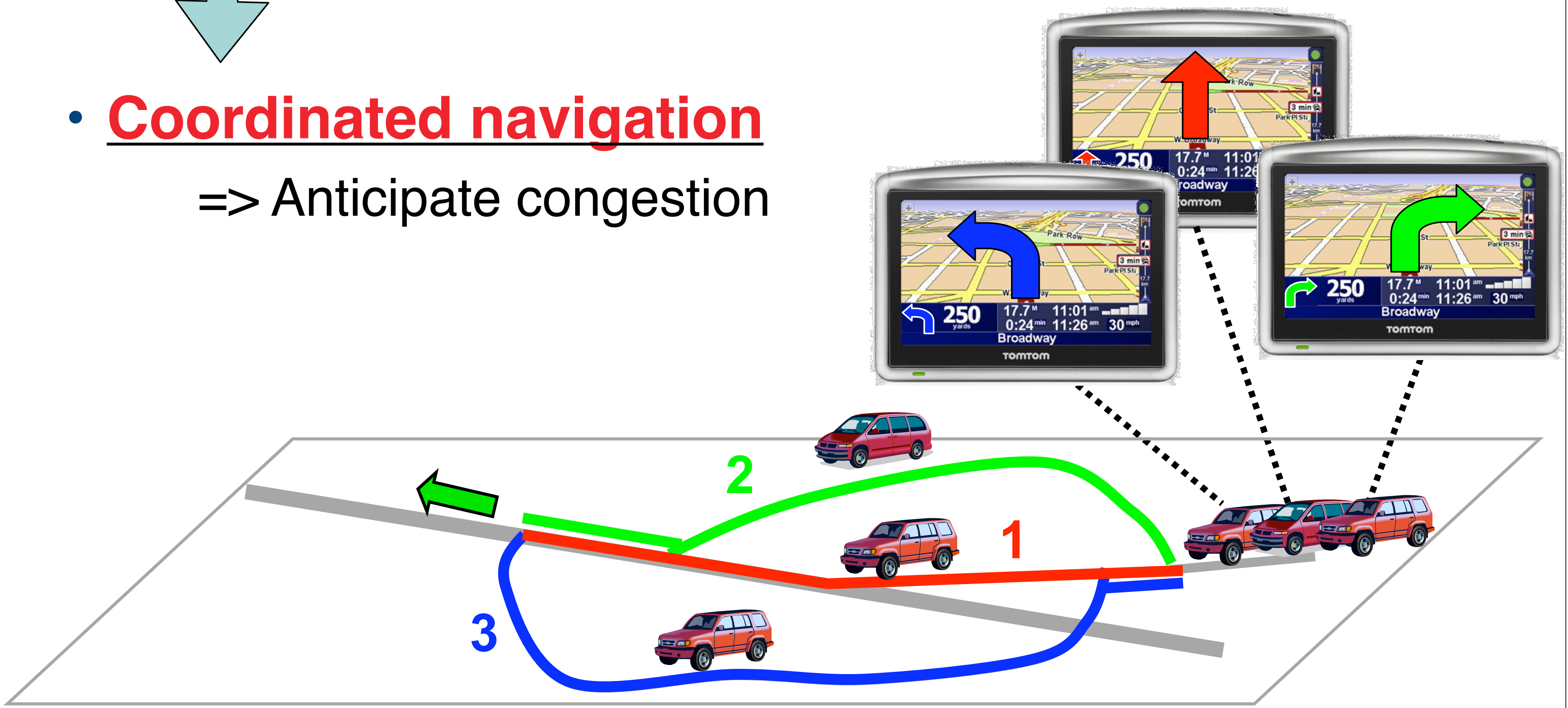
Anticipatory Vehicle Routing

Individual navigation

=> Congestion

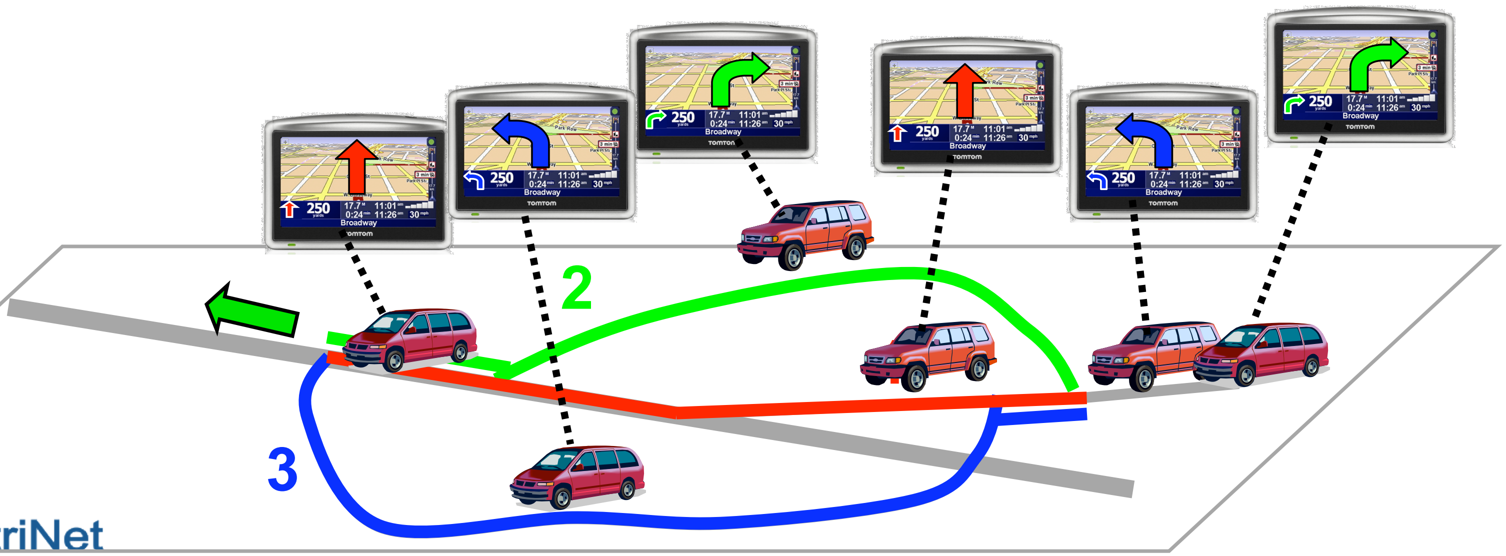
- **Coordinated navigation**

  => Anticipate congestion



Anticipatory Vehicle Routing

- Large-scale coordination problem...
- ... in a changing environment
  - Traffic jams come and go
  - Cars/drivers come and go



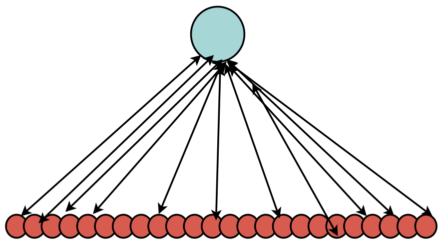
Coordination and Control Applications

- family of applications, characterized by
  - control application
    - underlying (physical or software) system that needs to be controlled
      - resources – static entities
      - mobile entities
    - on top: software system to "control" the underlying system
    - different order of magnitude of evolution speed
  - task-oriented application domain
    - a task entails moving through the environment (mobile entities) and performing operations using resources (static entities)
  - large / huge scale
    - number of entities
    - physical distribution
  - very dynamic nature
    - resources / connections / tasks

- examples
  - traffic control
  - AGV-based warehouse management
  - inland shipping
  - manufacturing control
  - supply chaing mgt
  - web service coordination

Solution?

- **Centralized appraoches**
  → consider the problem to be an optimisation problem
  → operations research / static and dynamic
  → feasibility ...?



- ...

Solution?

- **Centralized appraoches**

  → ...

- **Distributed approaches**

  → local decision makers, which cooperate / coordinate...

  - vehicles / roads
  - traffic lights / ...

  → crucial problem remains: deal with scale and complexity

Solution?

- **Distributed approaches**

  → crucial problem remains: deal with scale and complexity

  → compromises ...

  - hierarchical models

    - e.g. based on geographical characteristics...

    - compromises on flexibility, performance, complexity
    - e.g. 2-level distribution... [Klaus Fischer ’95]

  - pure decentralization
    - simple local rules + rely on emergence
    - compromise on optimality [Tamas Mahr ’08]

- **What is at the heart of the problem...**

→ **local** decision makers

→ require **global** information for adequate decision making

- **What could be at the heart of the solution...**

→ **<u>local</u>** decision makers

→ find/isolate only that **<u>global</u>** information **that is directly relevant** for adequate decision making

KATHOLIEKE UNIVERSITEIT LEUVEN DistriNet RESEARCH GROUP

Delegate MAS

- **Reference model** for Coordination and Control applications

→ Decentralized components / agents

→ Environment-centric coordination model

Basic elements in the architecture...

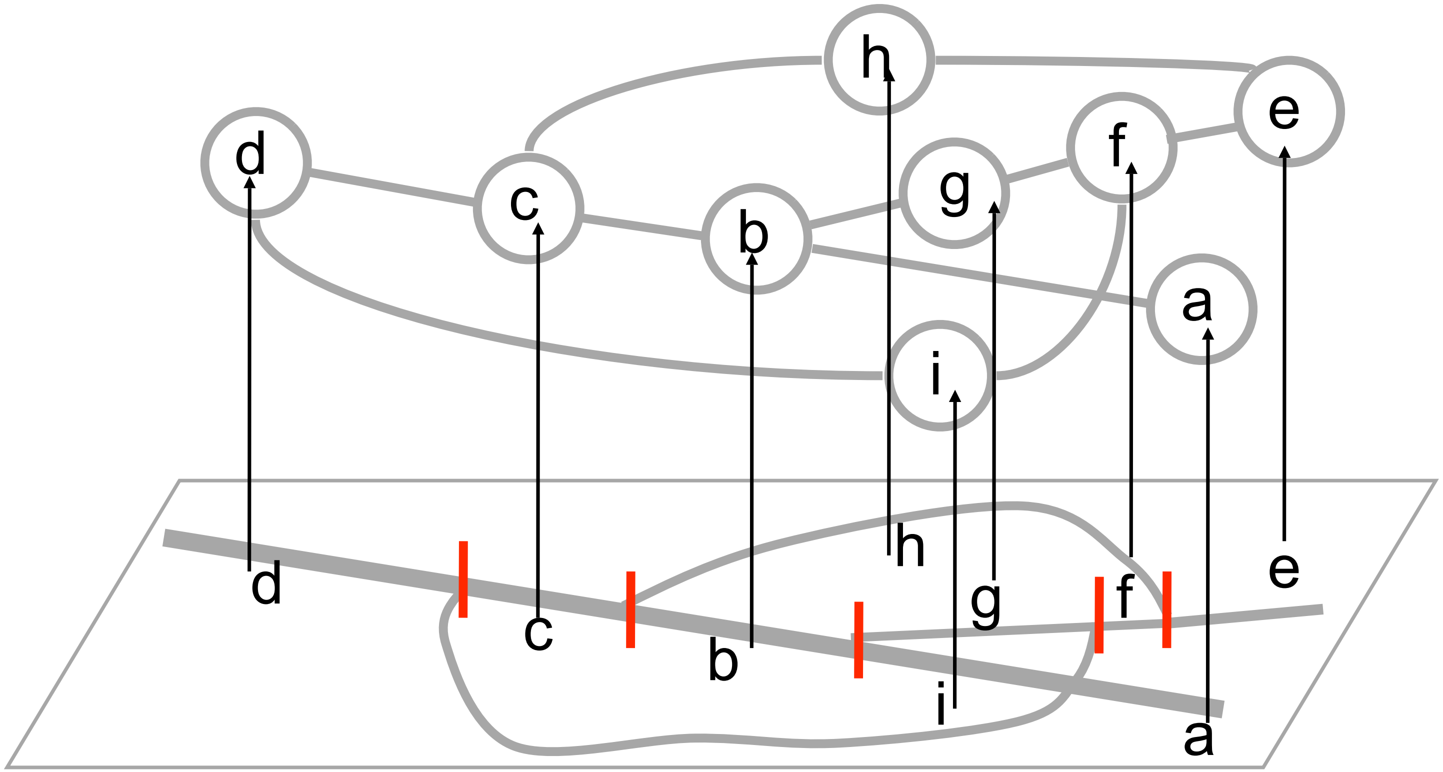
- Agent types
  → Resource / infrastructure agent
  → Task / vehicle agent

- Environment

- Coordination model

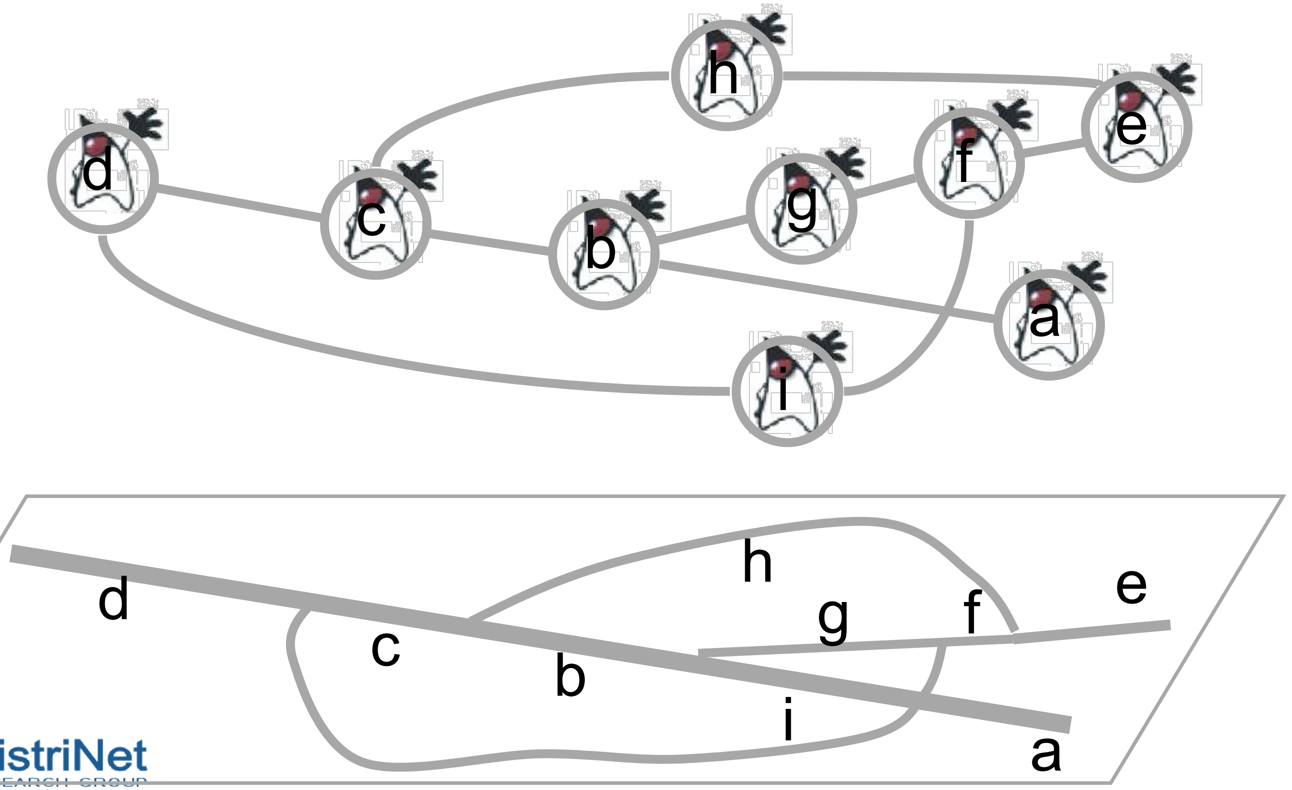
Virtual Environment

- Represents layout of road network
  - Roads divided in segments: graph



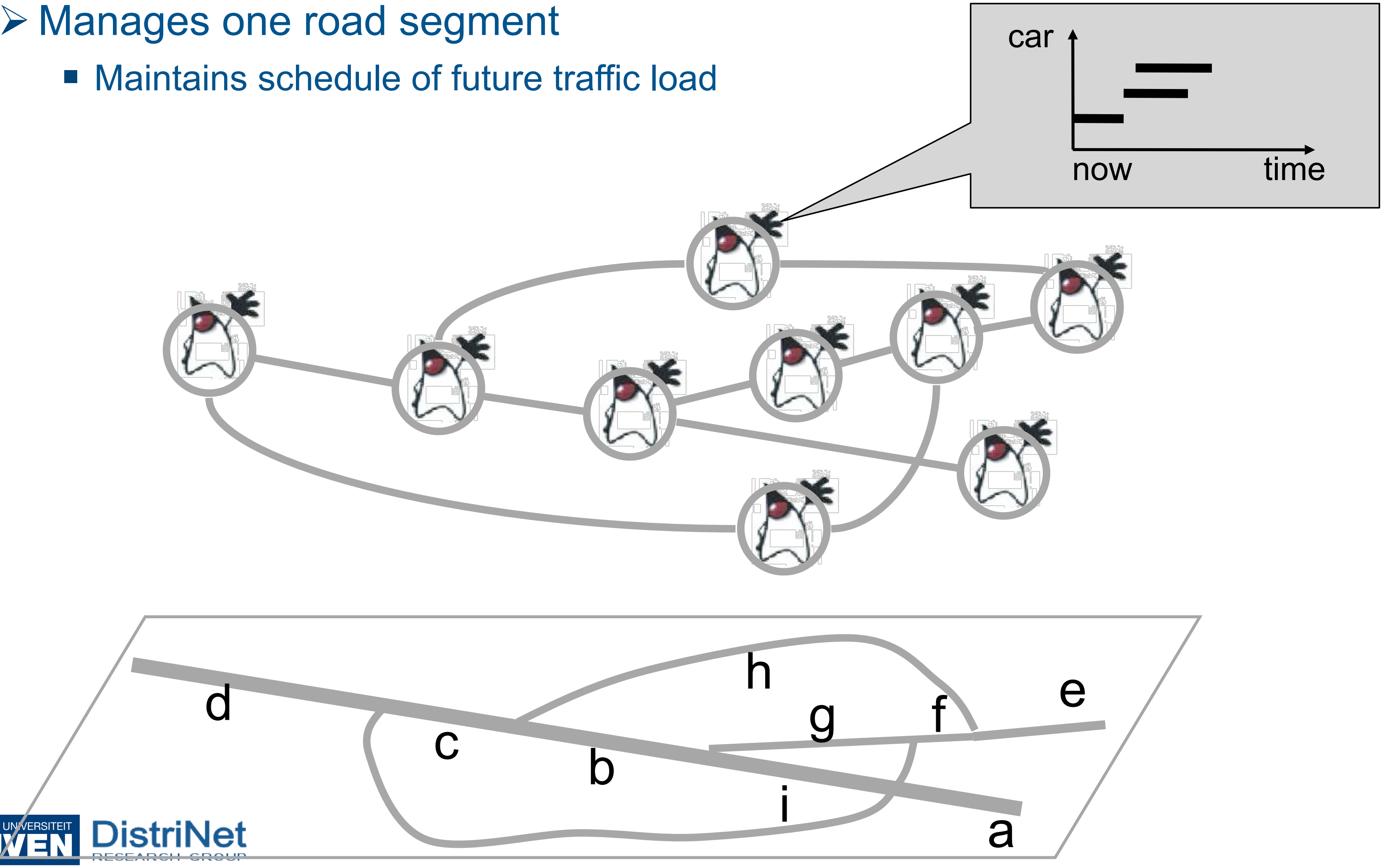
Infrastructure Agent

- Manages one road segment



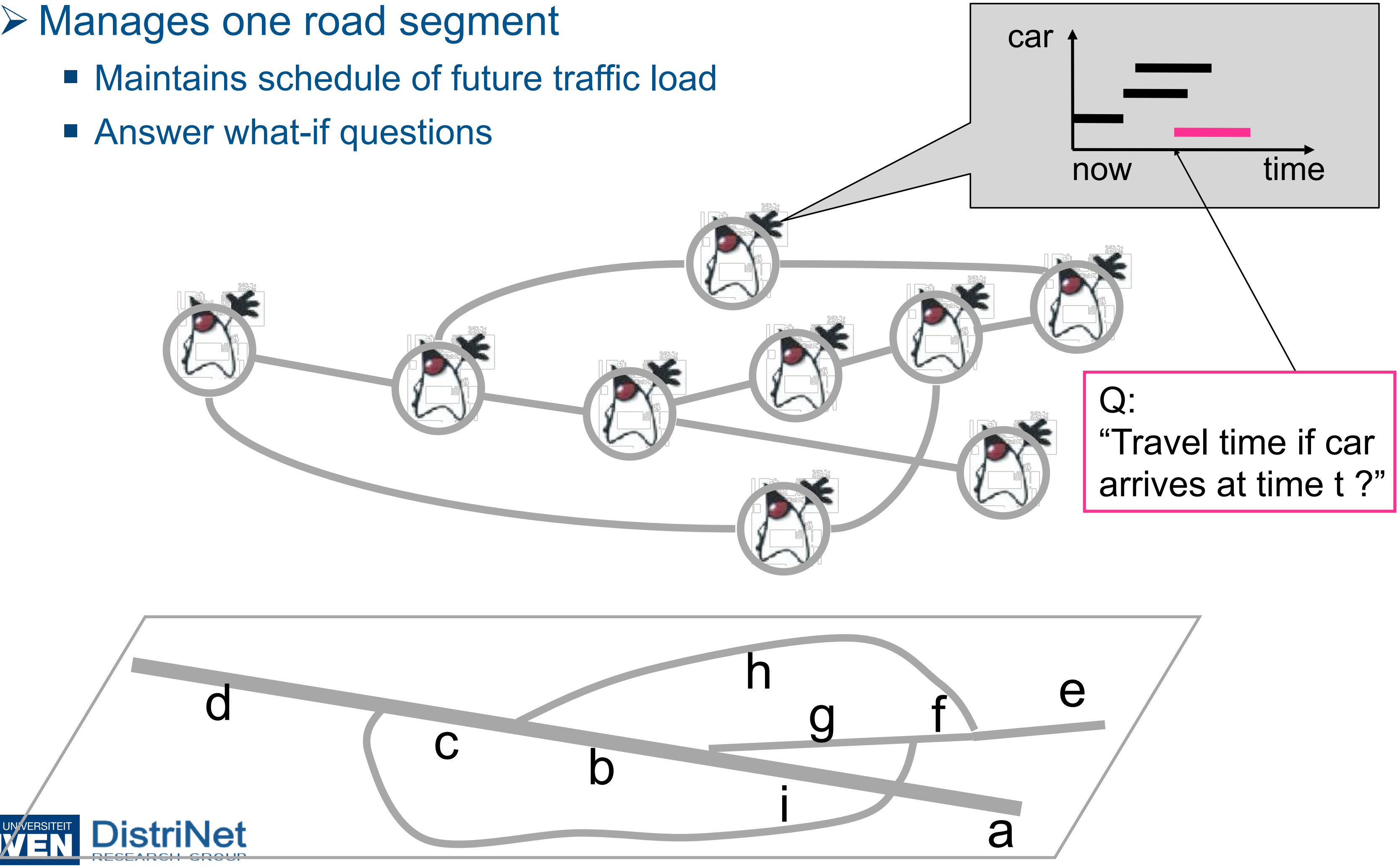
Infrastructure Agent (cont.)

- Manages one road segment
  - Maintains schedule of future traffic load



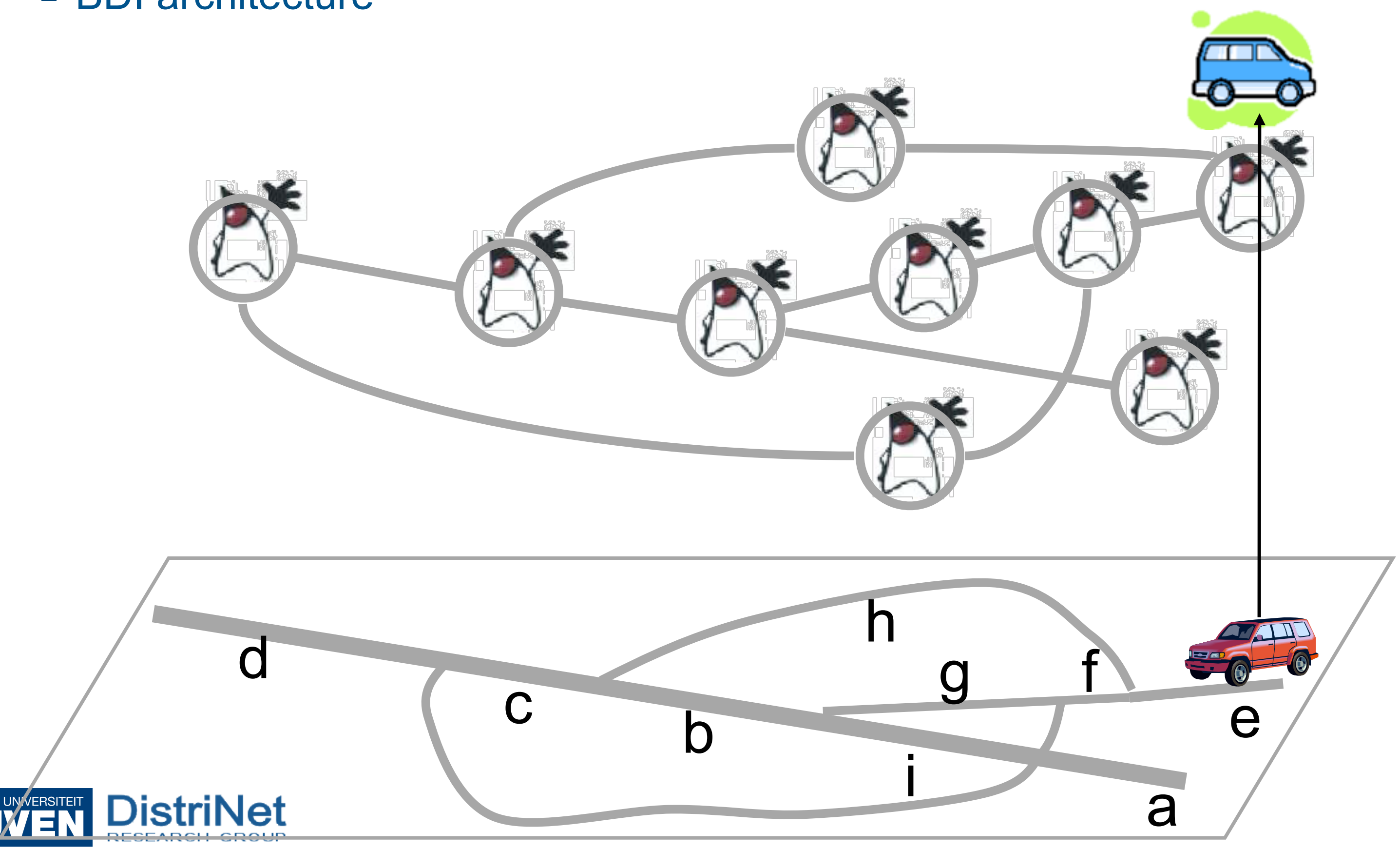
Infrastructure Agent (cont.)

- Manages one road segment
  - Maintains schedule of future traffic load
  - Answer what-if questions



Vehicle Agent

- Represents one car
  - Directs navigation of car
  - BDI architecture



Vehicle Agent (cont.)

- **Vehicle agent**
  - → overall “goal”:
    - fulfill task
    - by moving over resources in some correct sequence
    - fulfilling expected timing and quality requirements of the task
  - → BDI – Beliefs–Desires–Intentions
    - beliefs
      - resources
      - possible (feasible) paths for reaching resources
      - other task agents ?
    - desires / options
      - several paths through the infrastructure / resources
    - intention
      - a chosen path

Coordination model

- Basic entities in place
  - → environment
  - → infrastructure agents
  - → vehicle agents

- Now the system should support agents fulfilling tasks
  - tasks are trips to destinations
- ...taking timing and quality requirements into account...
  - minimize travel time
  - avoid traffic jams
  - ...

→ all this in an environment that changes constantly …
and in which task agents enter the system constantly …

Task agents

→ how ? first alternative:

task/vehicle agent responsible for gathering, reasoning upon and distributing information

- about resources / roads
  - topology
  - capabilities / quality / …
  - expected schedule
- about paths
  - find out feasible routes
  - contact resources on paths judge on the quality of these paths
- about intentions
  - communicate own intentions to other agents
  - negotiate with other agents to align all agents' intentions
  - reserve resources if suitable

complex

Task agents – BDI ?

→ how ? first alternative:

task/vehicle agent responsible for gathering, reasoning upon and distributing information

- all this in an environment that changes constantly … and in which task agents enter the system constantly …

complex

KATHOLIEKE UNIVERSITEIT LEUVEN DistriNet RESEARCH GROUP

Task agents – BDI ?

→ how ? <u>second alternative</u>:

exploit environment to relief task/vehicle agents …

➔ **<u>delegate MAS</u>**

- have simple, small-scale agents (ants) roam environment and enrich enviroment with valuable information
  - optional paths
  - intentions
- align intention with intentions of other task agents
  - through resource agents
  - through refresh

Delegate MAS: Ant-based Coordination and Control

→ three kinds of delegate MASs

- Exploration ants
- Intention ants
- *(Feasibility ants)*

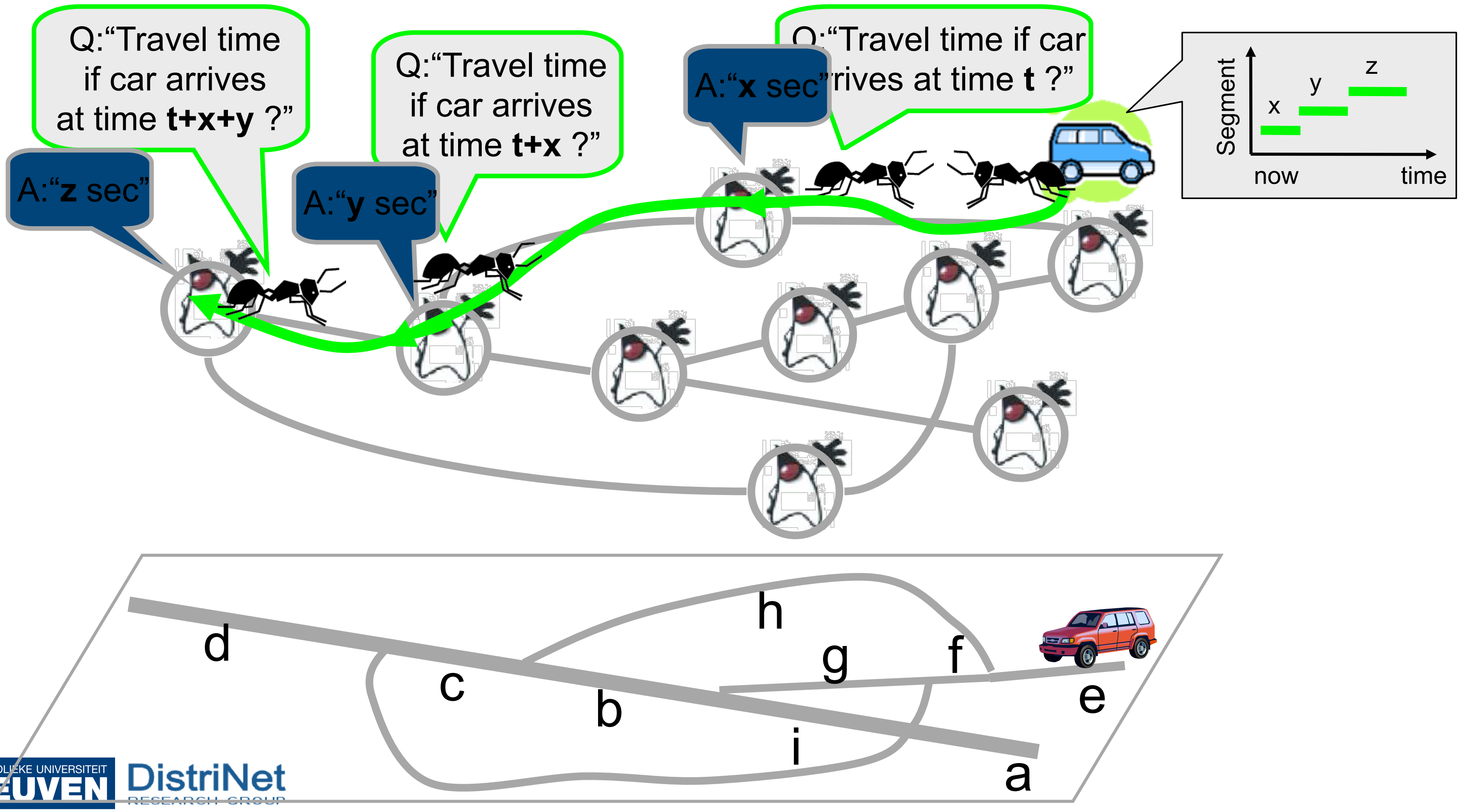
1. Ant agents
2. Pheromone deposition spaces

   attached to each resource/enty/exit

   ! evaporation and refresh

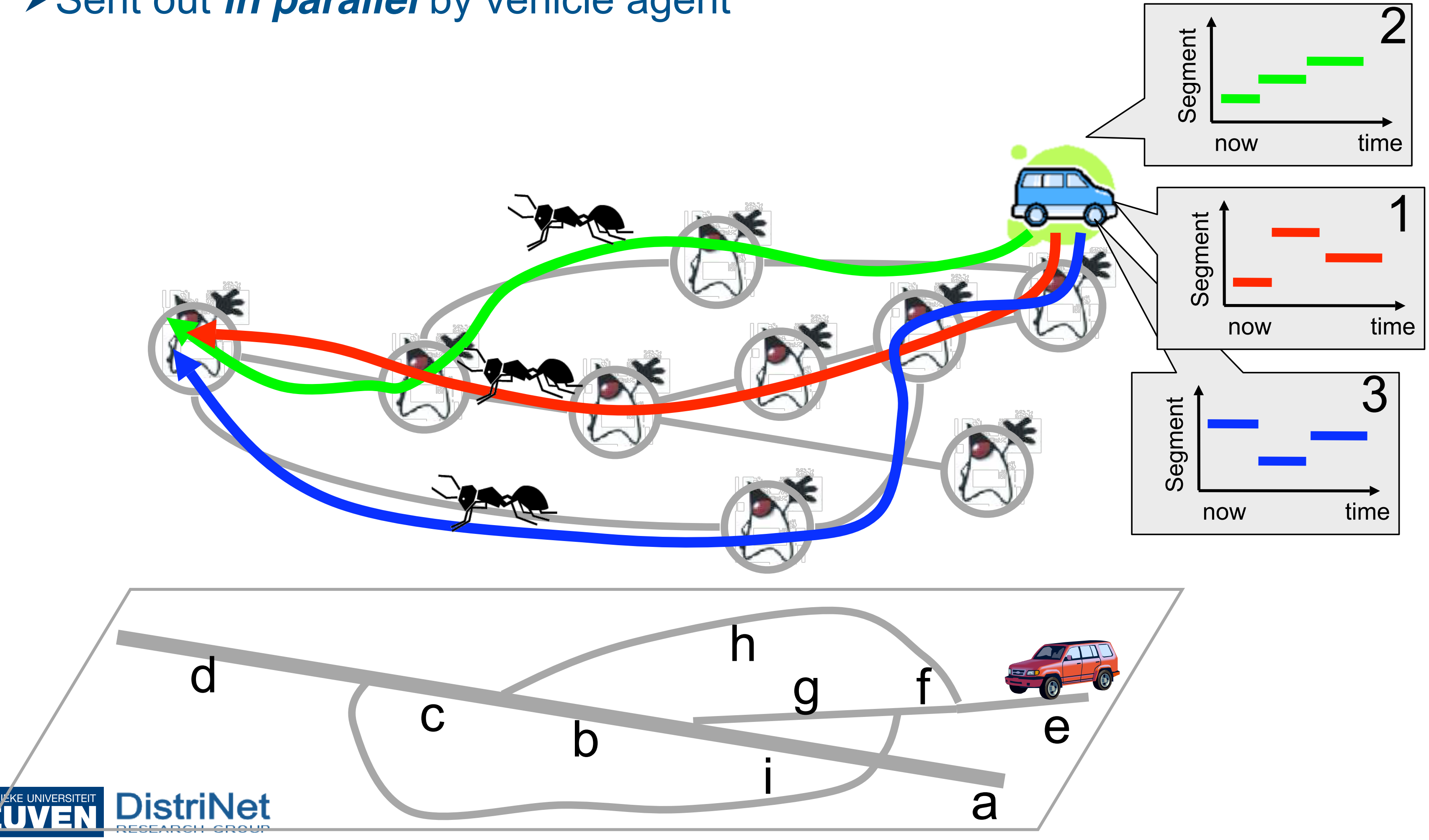
Exploration Ants

- Explore quality of alternative routes @ cyber speed



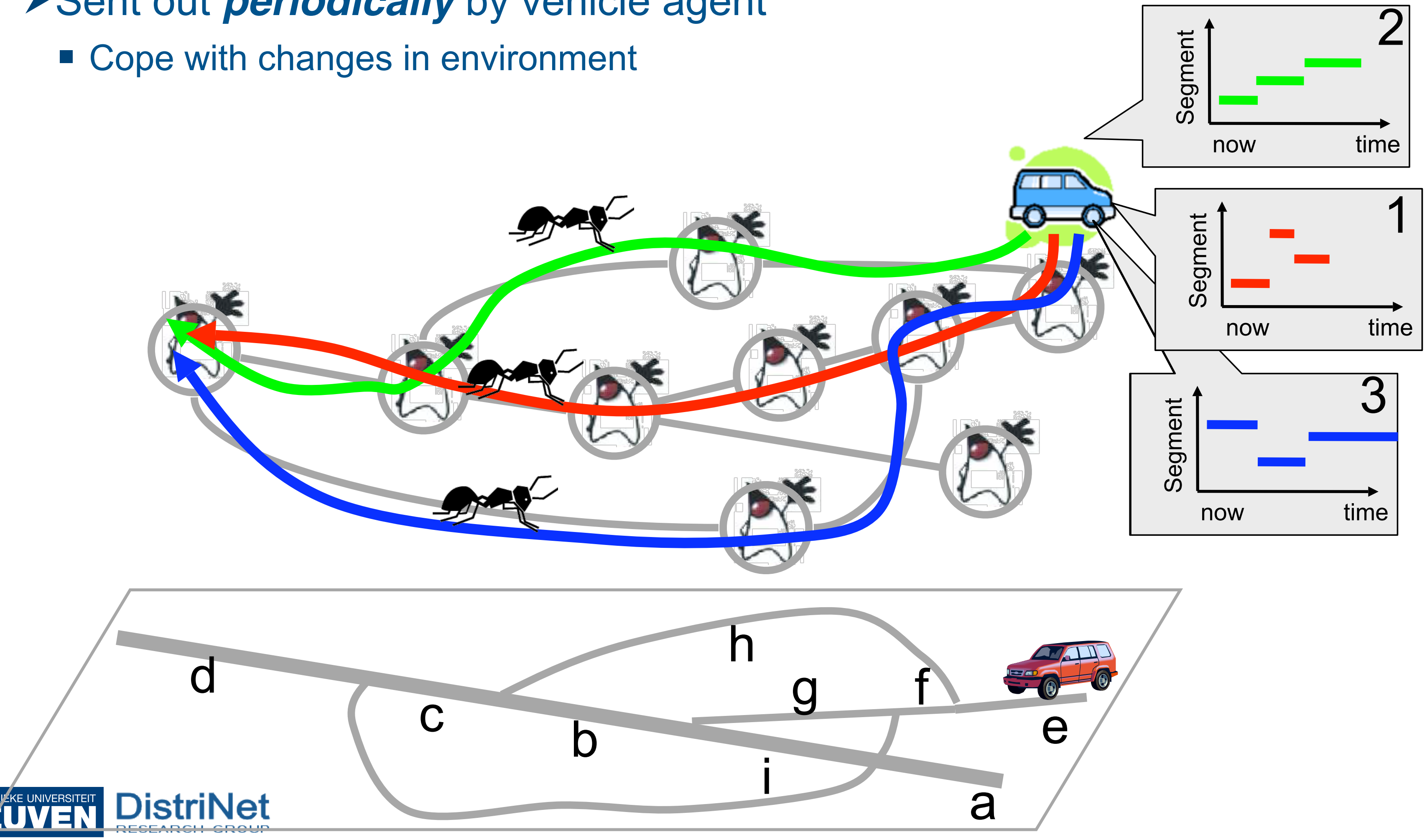
Exploration Ants

- ➢Sent out ***in parallel*** by vehicle agent



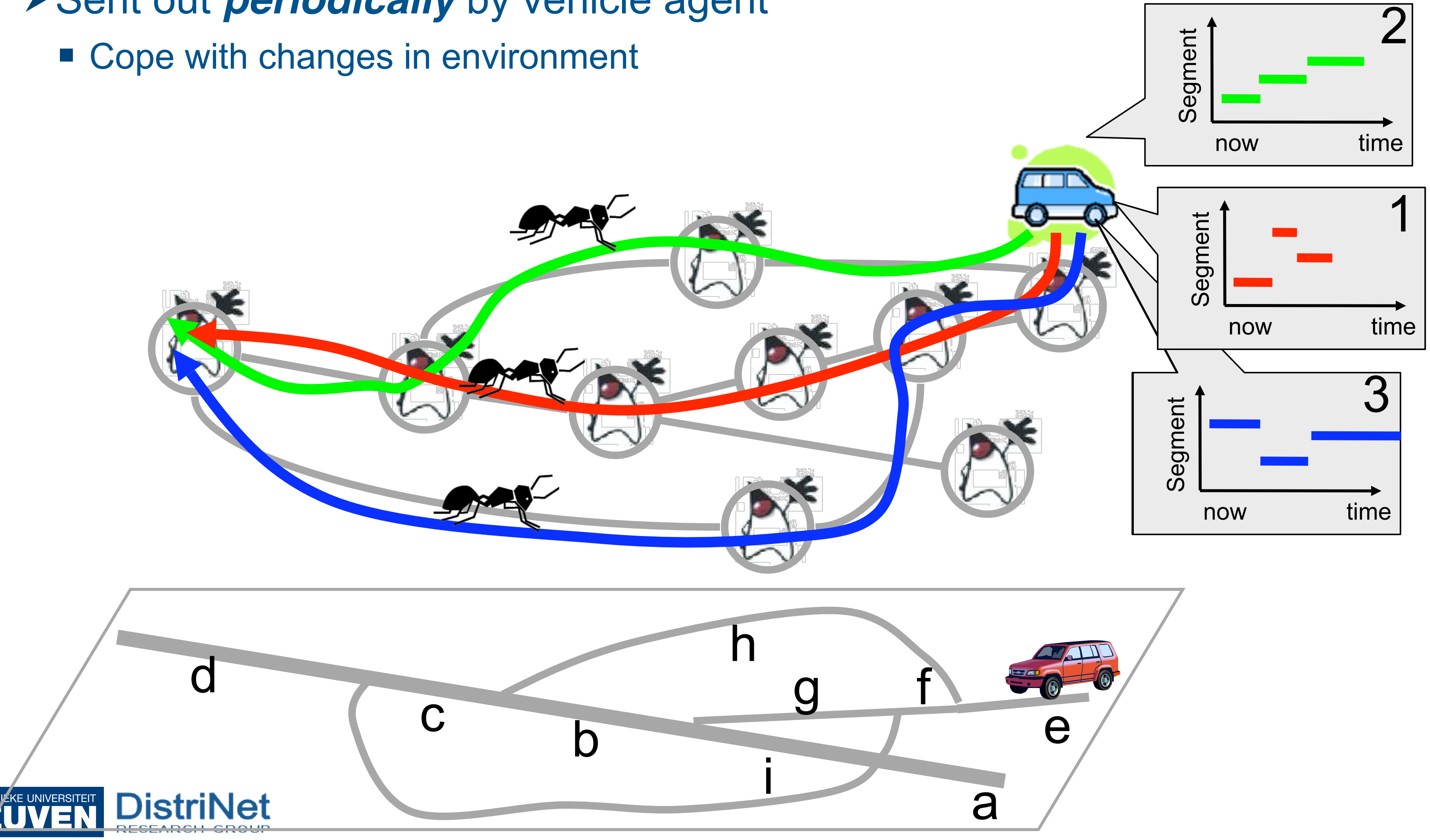
Exploration Ants

- Sent out ***periodically*** by vehicle agent
  - Cope with changes in environment



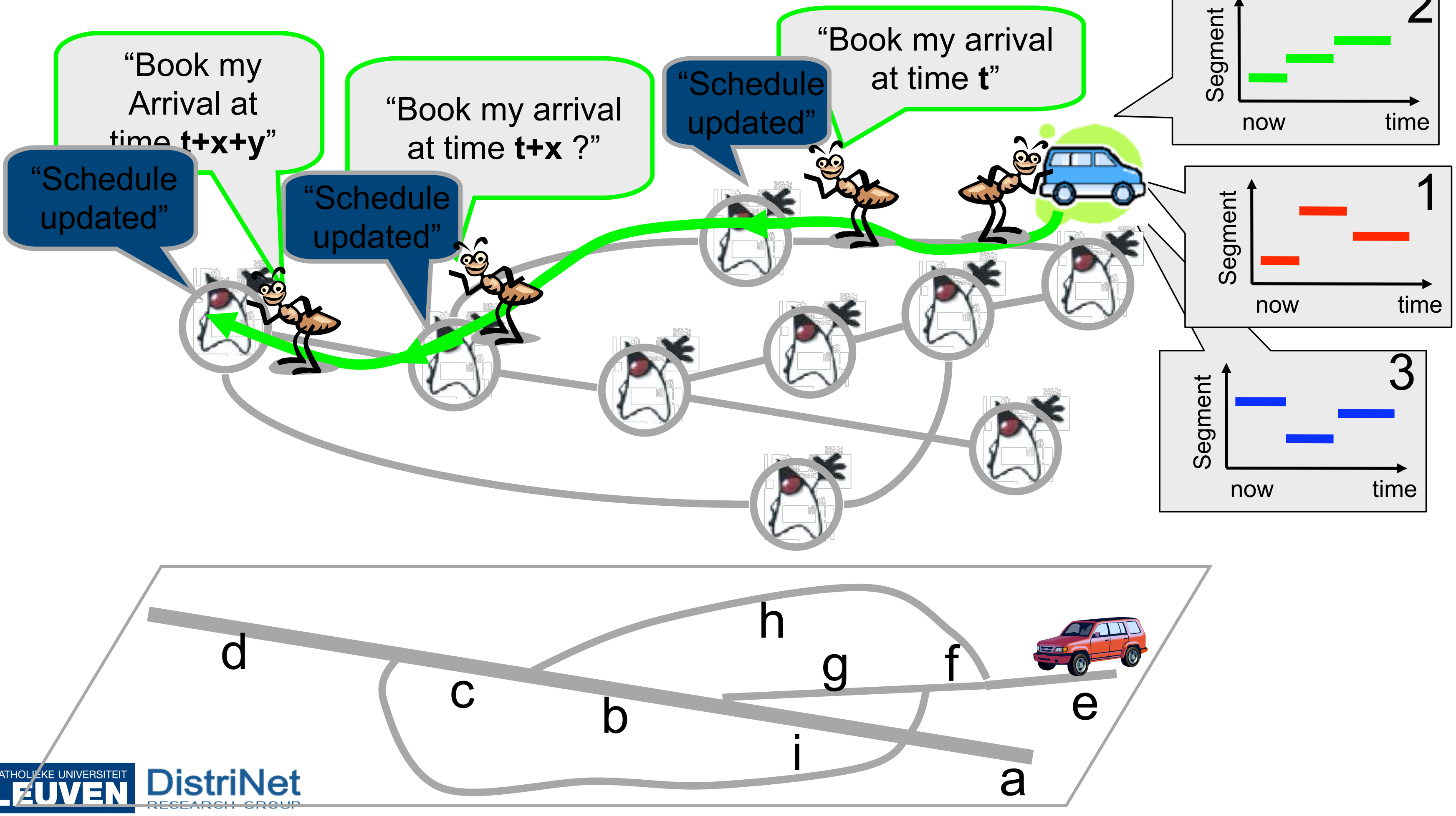
Exploration Ants

- Sent out ***periodically*** by vehicle agent
  - Cope with changes in environment



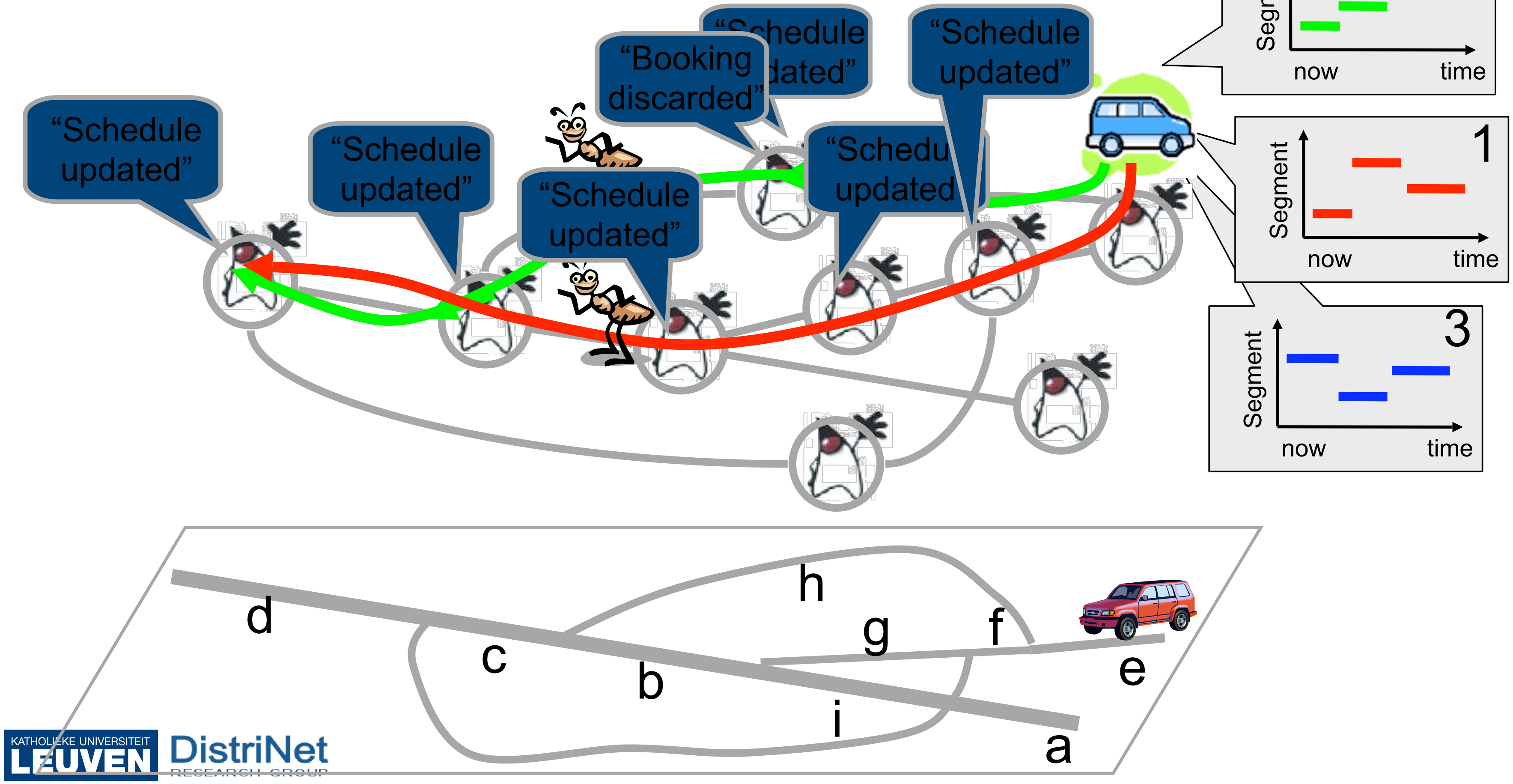
Intention Ants

- Tentative booking of intended route

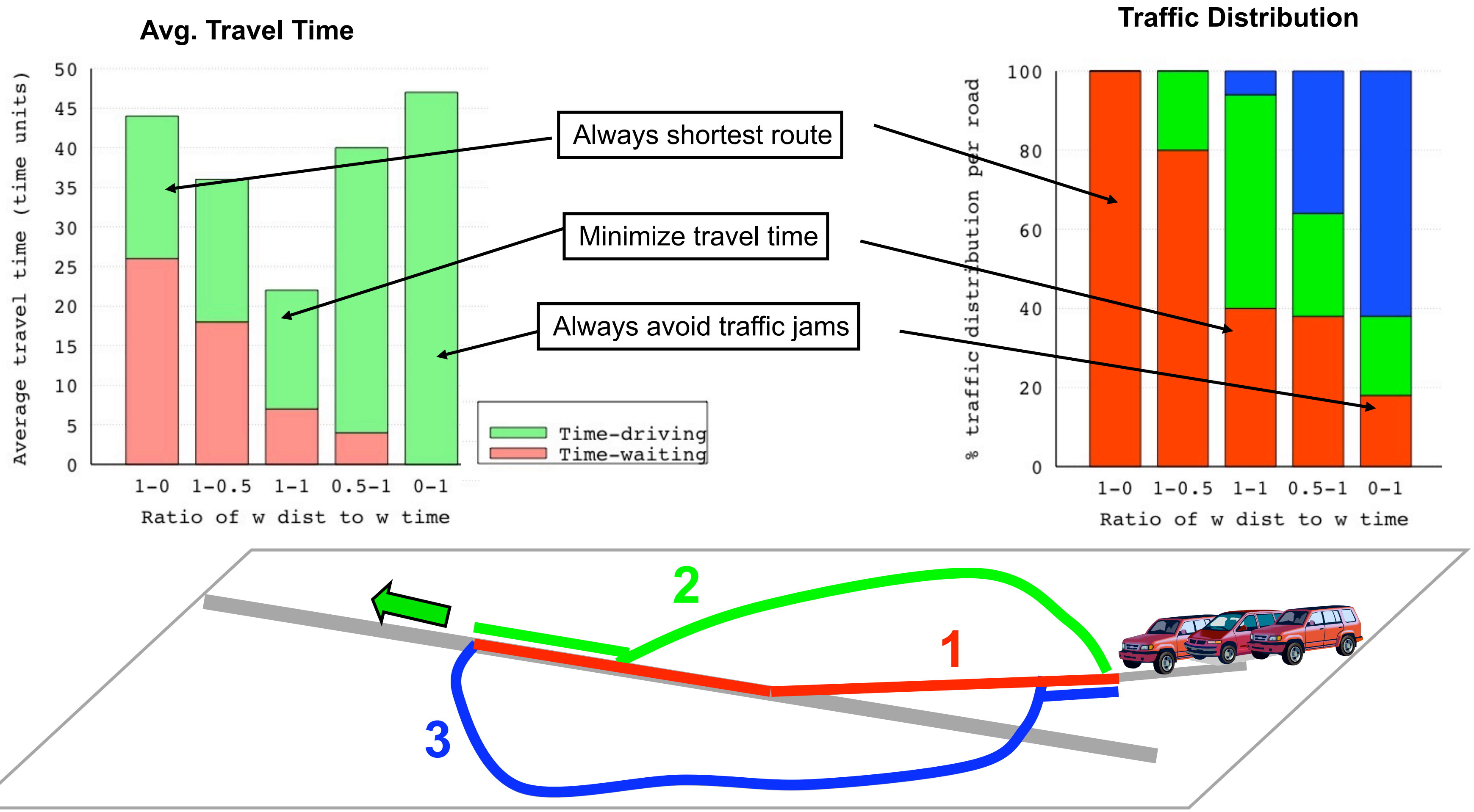


Intention Ants

- Bookings decay => rebook periodically
  - Cope with intention revision



Prototype: Experiments



Conclusion

- Delegate MAS reference model
  - → core abstractions
    - environment
    - task/vehicle agents - basically BDI
    - resource/infrastructure agents
  - → coordination model
    - environment centric
    - light-weight ‘ants’ + pheromones
    - bring relevant global information to local task agents
      spread relevant global information through environment
  - → cope with dynamics
- ...

Conclusion (cont.)

- Exploration delegate MAS
  - use the environment to find out the quality of different options

⇨ *task agents do not need to directly contact and negotiate with resources*

- Intention delegate MAS
  - use the environment to propagate intentions through the environment

⇨ *task agents do not need to maintain beliefs and reason upon the intentions of other agents' for coordinating over resources*

➔ reduced complexity of task agent architecture

Many challenges / Open Issues

- There's a **cost** …
  - additional infrastructure
  - open resources
  - computational/communication cost
    - needs to be managed properly!

      suitable refresh rate, cloning budgets, hop limits
- **Emergent behaviour / qualities**

  → … ?

  → purely selfish agents – sufficient for overall optimization ??

  → homogeneous or heterogeneous?
- **Many parameters**

  → tune ? adaptive strategy ?
- **Coordination between resources**
- **BDI architecture**
- **Resource agent architecture**
- …

Current / Future Work

- Evaluation / Validation
  - → MASE project - large scale microscopic traffic simulations
  - → Float line cold end
  - → Inland shipping?
  - → PDP-TW

Interested?
There is a **vacancy**...

Using Equation-Free Macroscopic Analysis for Studying Self-Organising Emergent Solutions

SASO'2008

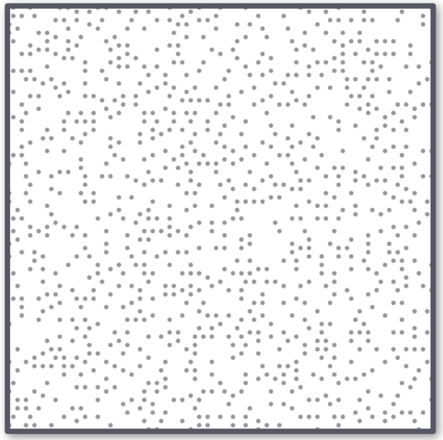
Giovanni Samaey – Numerical analysis group
Tom Holvoet – DistriNet labs
Tom De Wolf – DistriNet labs

Overview

- Problem: how to evaluate SO-em solutions
  → Illustrated on a case
- Iterative approach to studying SO-em solutions
  → Principle
  → Equation-free analysis
  → Illustration / case
  → Discussion / challenges
- Conclusion

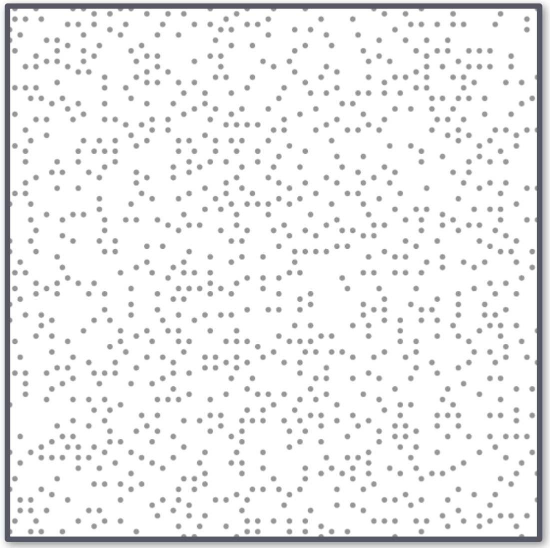
A simple case: data clustering

- Requirements
  → Functional: clustering data in a 2D grid
  → Non-functional: open, …

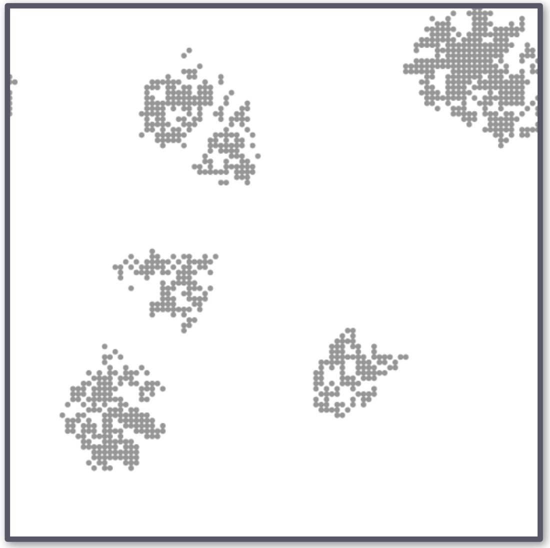


- Architecture
  → MAS: 2D grid world, inhabited by autonomous agents
  → Agents follow ants' brood sorting behaviour
    - Simple rules:
      - Agents pick up data and drop it (with probability P) if in neighbourhood (8 units view range) of other data items (3)
    - Self-organizing emergent solution

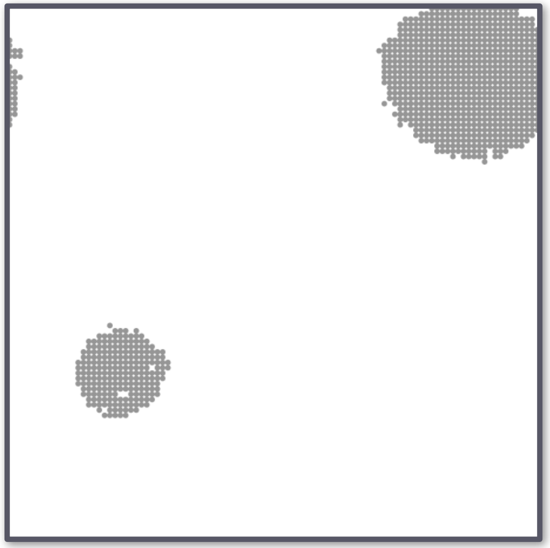
- Decentralized data clustering…



Step 0



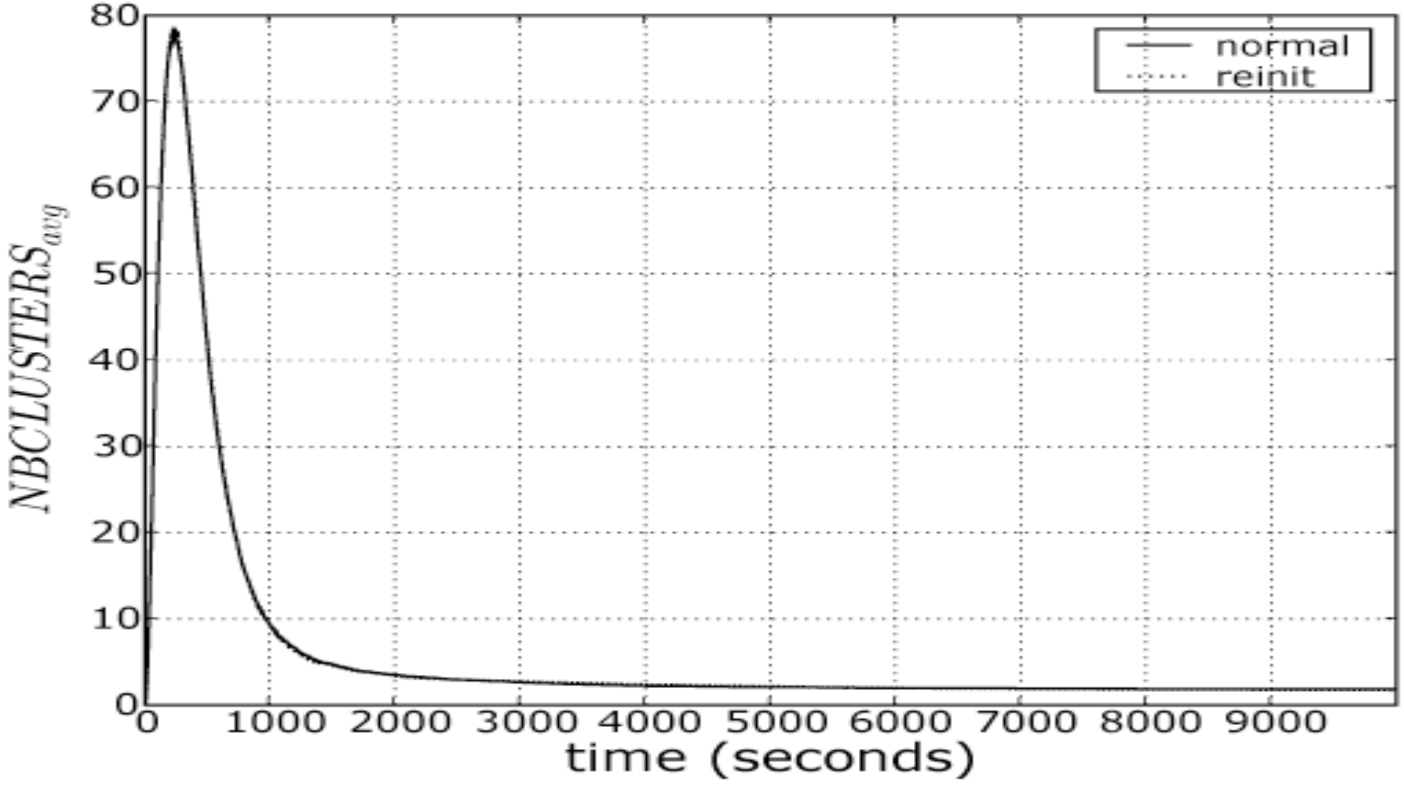
Step 1000



Step 9000

KATHOLIEKE UNIVERSITEIT LEUVEN DistriNet RESEARCH GROUP

- Decentralized data clustering…
  - → Evolution of the number of clusters
    - Avged over 100 simulations from random initial conditions



Decentralized data clustering…

- Avg cluster size is 1.6…
  → How come?
  → Can we fix the solution to get better results?

- Now what?
  → We need to better understand how this solution works…
  → Where does the global behaviour come from?
  → How do local actions lead to this global behaviour?

Problem

- How to evaluate a SO-em solution?

Problem

- How to evaluate a SO-em solution?

→ If you can derive a mathematical model analytically: OK!

  - In many, real ‘engineered’ systems, you cannot derive such a model…

Problem

- How to evaluate a SO-em solution?

→ If you can derive a mathematical model analytically: OK!

- In many, real 'engineered' systems, you cannot derive such a model…

→ Pure simulation

- Simulate ➔ what will you analyse?
  - Ok for *observing* what the macroscopic behaviour is…
  - Less ok for *evaluating* the solution.
  - How to proceed if results are not satisfactory?

→ …

Problem

- How to evaluate a SO-em solution?

→ If you can derive a mathematical model analytically: OK!

- In many, real 'engineered' systems, you cannot derive such a model…

→ Pure simulation

- Simulate ➔ what will you analyse?
  - Ok for *observing* what the macroscopic behaviour is…
  - Less ok for *evaluating* the solution.
  - How to proceed if results are not satisfactory?

→ We would like to have a disciplined approach
to grasp an understanding of SO-em solutions…
which can help make supported claims about macroscopic behaviour and which can help improve the solution

Solution: how to proceed…

- An iterative, bottom-up approach…
  → Based on a procedure that tries to identify, step by step, a minimal set of macroscopic variables ***U*** that completely and 'uniquely' characterise the SO-em behaviour
    - Necessary and sufficient set of macro-variables

  → Finding necessary and sufficient macro-variables is only a *tool*, not the *goal*
  → It may not be achievable, but it's about the ride, not about the destination

- The approach
  → … does *not tell* you *what* the micro-macro relation is
  → … but *helps* or *guides* you in your *study* to understand the relation

An iterative, bottom-up approach…

1. Start from U = u (U = macroscopic, u = microscopic state)
2. Gradually aggregate state → U
   → Define macroscopic variables by abstracting microscopic state
3. Check whether U is sufficient to accurately determine overall system evolution
4. If sufficient, step 2

Using EFA

Equation-free analysis *[I. Kevrekidis, 2003]*

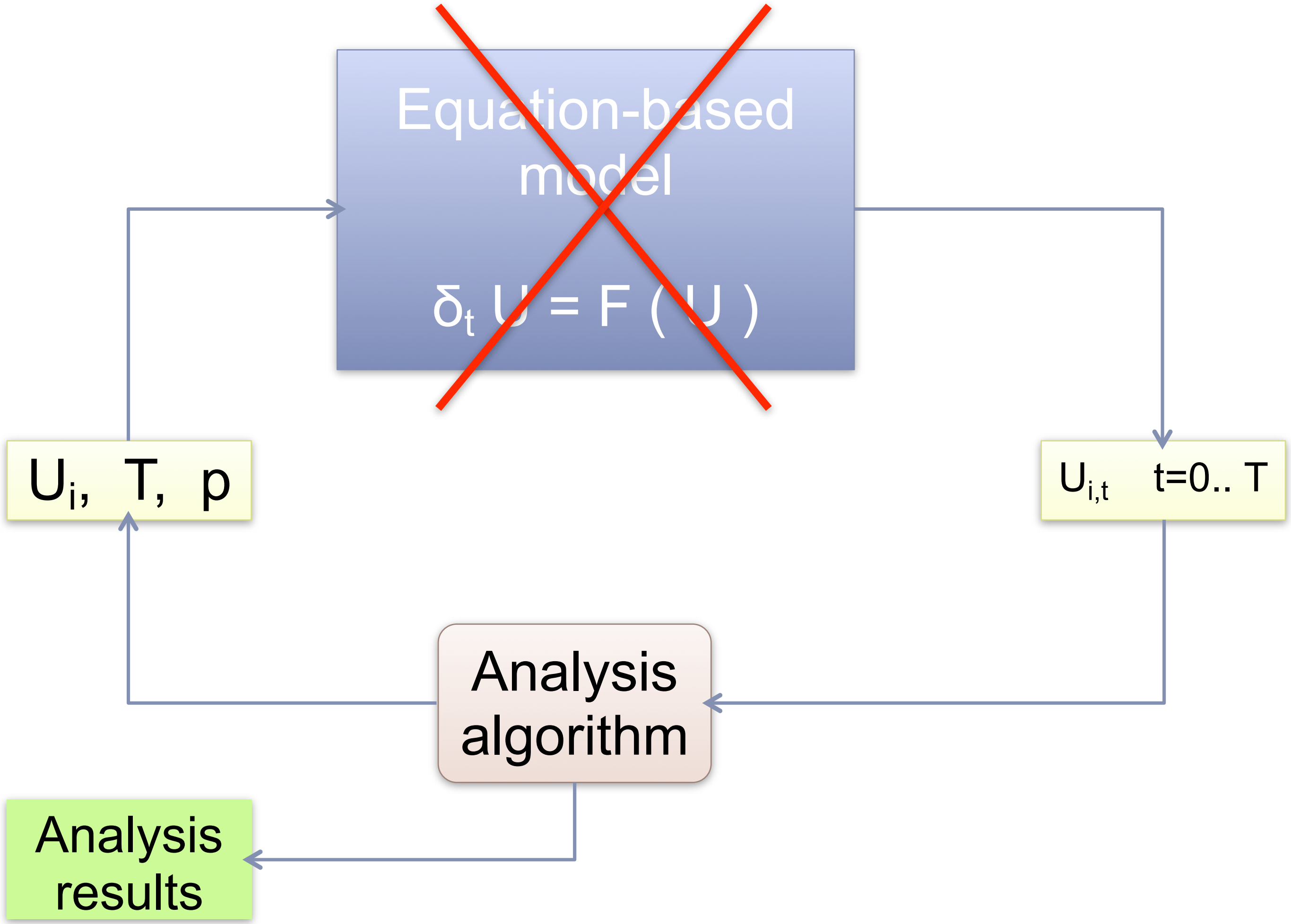
- Analysis if you *do* have an equation-based model in closed form:

$$\delta_t U = F (U) \qquad U\text{: macroscopic variables}$$

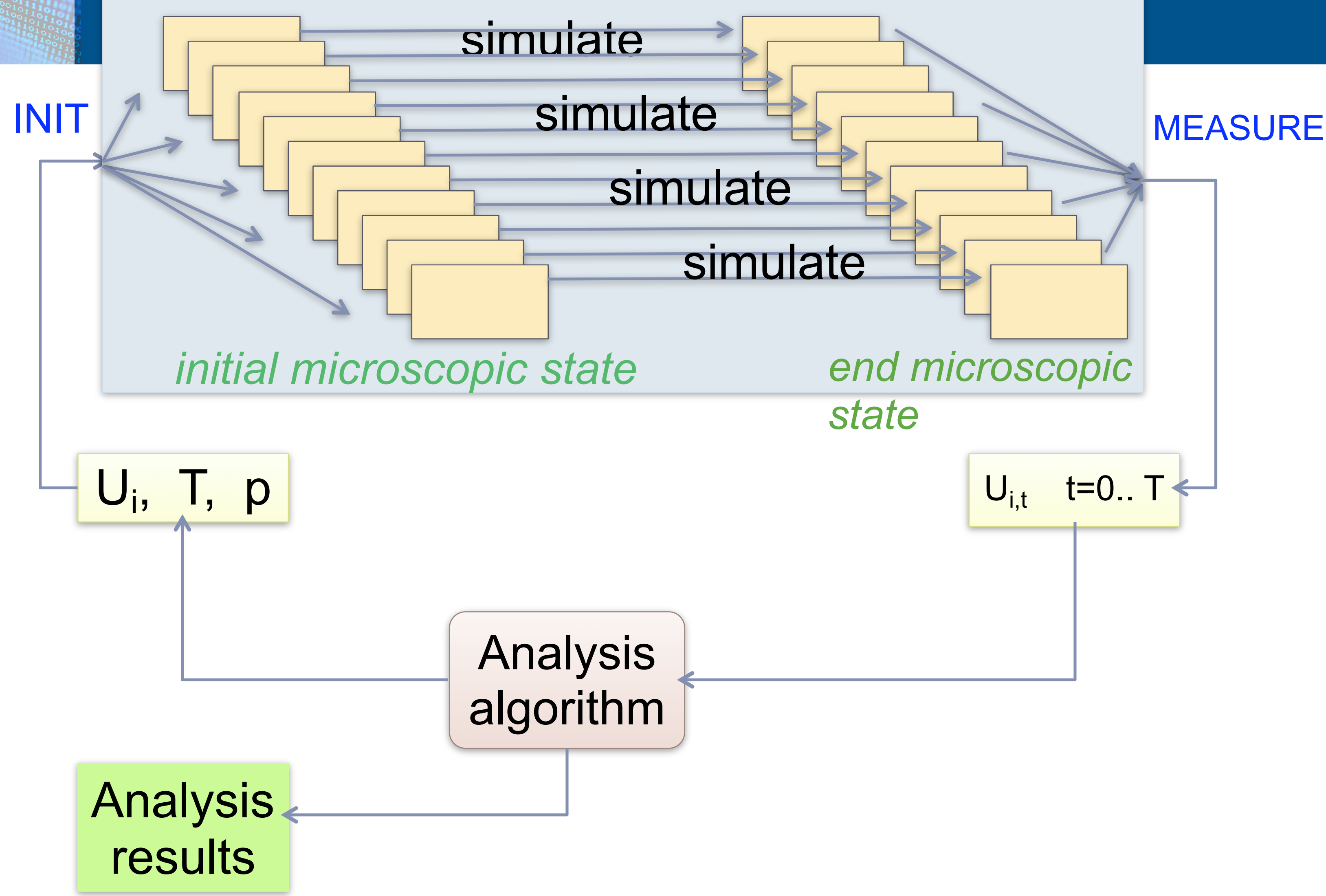
→ Analysis algorithms

- Require *evaluation* of the equation to obtain macroscopic values at certain time steps

$$U_i \rightarrow \quad \delta_t U = F (U) \quad \rightarrow \quad U_{i,\,t} \quad t = 0..T$$

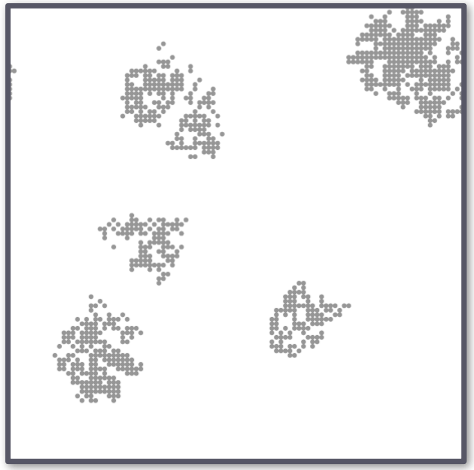

Equation-based model
δt U = F (U)
Ui, T, p
Ui,t t=0.. T
Analysis algorithm
Analysis results

Equation-free approach



An iterative approach… the data clustering example

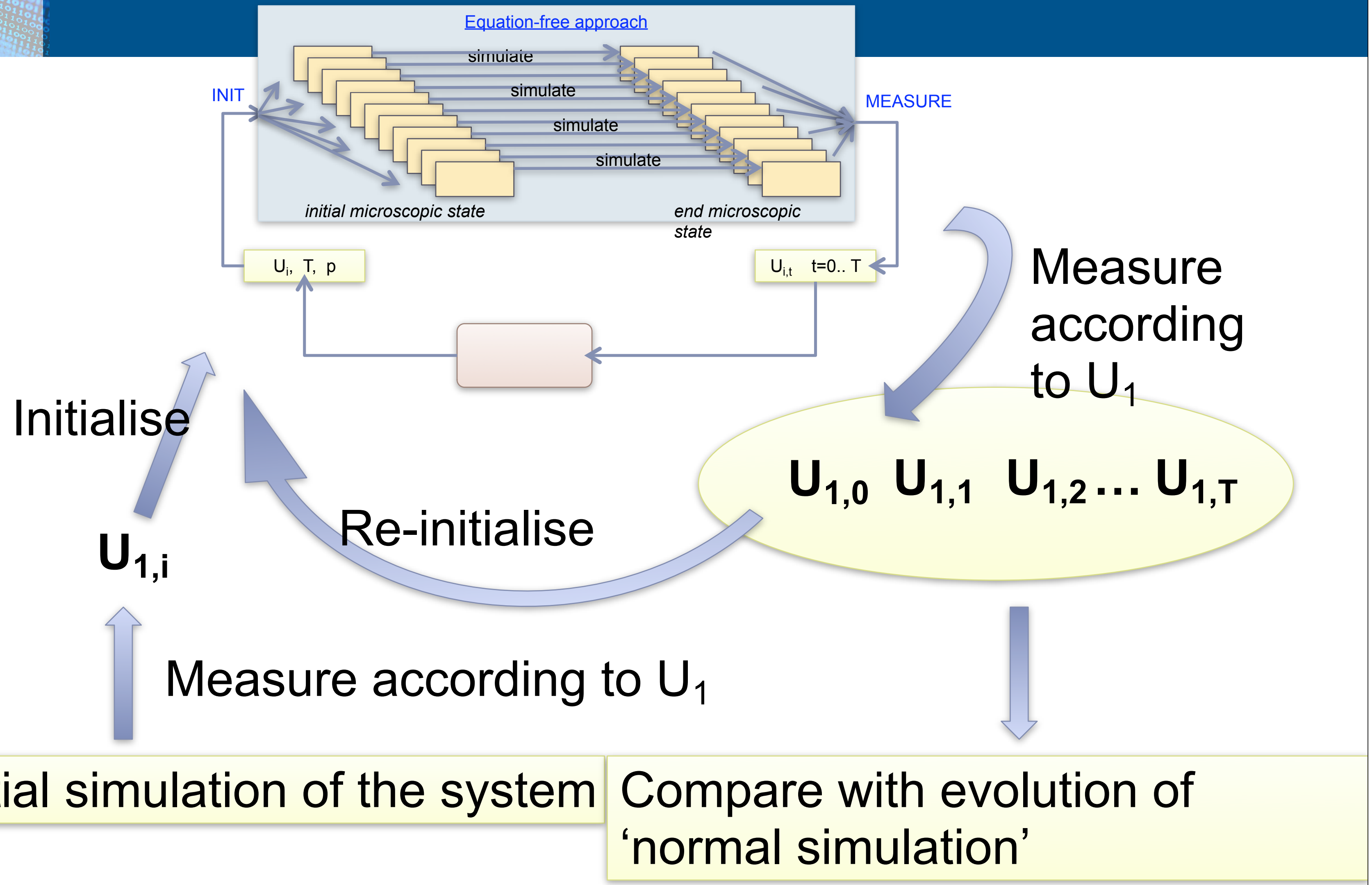
- Microscopic state:
  - → Data items: position
  - → Agents: position, direction, carrying data or not

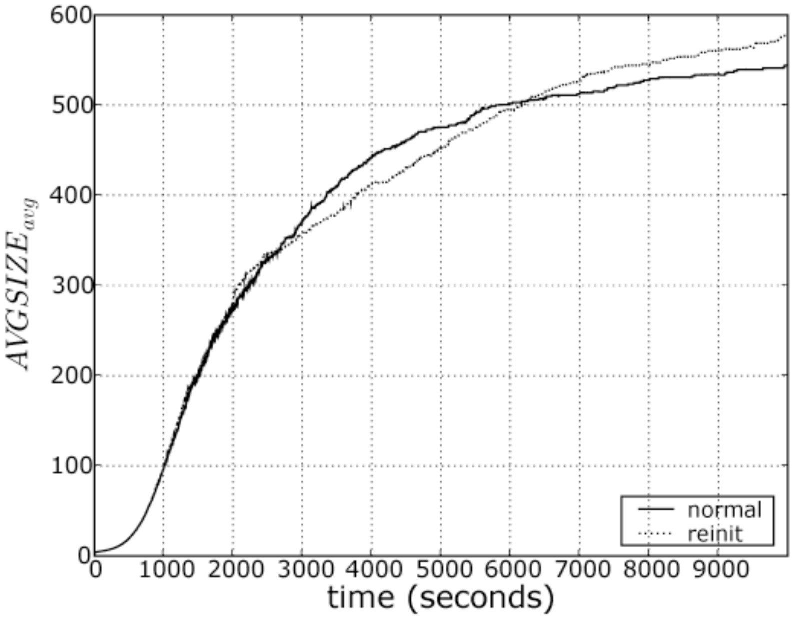


- Iteratively attempt/evaluate set of macro-variables + initialisation operator
  - → Macro-variables via aggregated state
  - → Initialisation operator fills degrees of freedom

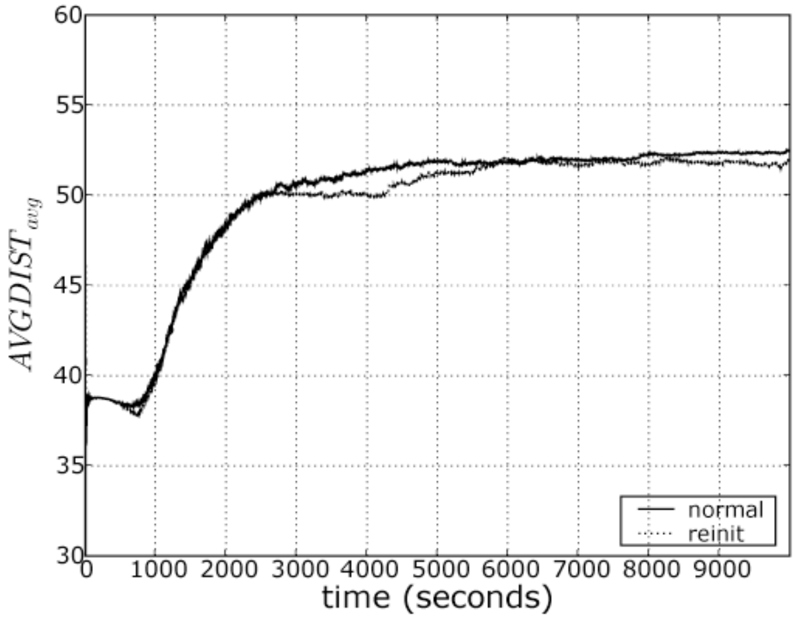
An iterative approach… initial attempt

- Initial attempt
  - → Aggregated state U_1
    - Clusters: number, position of center, size
    - Agents: number of agents carrying data
  - → INIT:
    - Clusters: clusters, same center position, exact positions of data items randomized
    - Agents: carrying → next to a data item, others random pos.


Equation-free approach
simulate
simulate
simulate
simulate
INIT
MEASURE
initial microscopic state
end microscopic state
U_i, T, p
$U_{i,t}$ t=0.. T
Measure according to U_1
Initialise
$U_{1,0}$ $U_{1,1}$ $U_{1,2}$ … $U_{1,T}$
Re-initialise
$U_{1,i}$
Measure according to U_1
Initial simulation of the system
Compare with evolution of ‘normal simulation’



600
500
400
300
200
100
0
$AVGSIZE_{avg}$
0 1000 2000 3000 4000 5000 6000 7000 8000 9000
time (seconds)
normal
reinit



60
55
50
45
40
35
30
$AVGDIST_{avg}$
0 1000 2000 3000 4000 5000 6000 7000 8000 9000
time (seconds)
normal
reinit

An iterative approach… initial attempt

- Initial attempt
  - → Aggregated state U_1
    - Clusters: number, position of center, size
    - Agents: number of agents carrying data
  - → INIT:
    - Clusters: clusters, same center position, exact positions of data items randomized
    - Agents: carrying → next to a data item, others random pos.

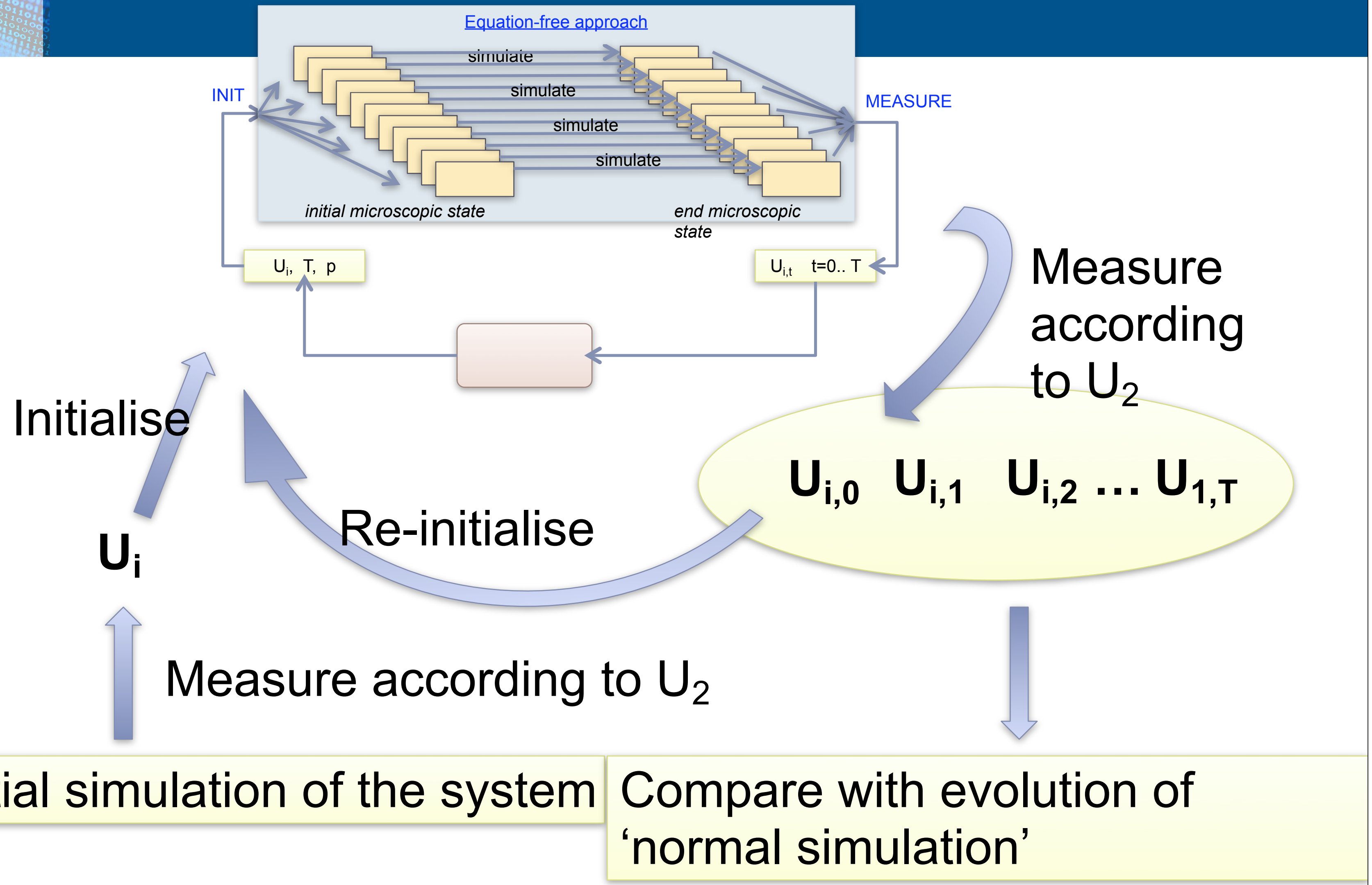
Evaluation:

The avg number of clusters decreases more slowly

U is IN-sufficient to accurately determine overall system evolution

An iterative approach… second attempt

- Second attempt
  - → Aggregated state
    - Clusters: number
position of center
size
    - Agents: number of agents carrying data
  - → INIT:
    - Clusters: clusters, same center position, **circle shaped, 10% vacant positions**, exact positions of data items randomized
    - Agents: carrying → next to a data item, others random pos.



Equation-free approach
simulate
simulate
simulate
simulate
INIT
MEASURE
initial microscopic state
end microscopic state
U_i, T, p
$U_{i,t}$ t=0.. T
Measure according to U_2
Initialise
$U_{i,0}$ $U_{i,1}$ $U_{i,2}$ … $U_{1,T}$
Re-initialise
U_i
Measure according to U_2
Initial simulation of the system
Compare with evolution of 'normal simulation'

An iterative approach… second attempt

- Second attempt
  - → Aggregated state
    - Clusters: number
      position of center
      size
    - Agents: number of agents carrying data
  - → INIT:
    - Clusters: clusters, same center position, **circle shaped, 10% vacant positions**,
      exact positions of data items randomized
    - Agents: carrying → next to a data item, others random pos.

Evaluation:

A periodic reinitialisation (every 1000 steps) did not alter macroscopic evolution
So: all macro-behaviour is contained in our initial set of macro-variables
U is sufficient to accurately determine overall system evolution

An iterative approach… second attempt

<u>Gained insight</u>:

the number of clusters decreases faster if there are *vacant places* in the clusters and if the clusters are *circular*.

An iterative approach… third attempt

- Third attempt
  - → Aggregated state
    - Clusters: number
      ~~position of center~~ → avg distance
      size
    - Agents: ~~number of agents carrying data~~
  - → INIT:
    - Clusters: clusters, positions according to avg distance, exact positions of data items randomized
    - Agents: randomized positions

Evaluation:

A periodic reinitialisation (every 1000 steps) did not alter macroscopic evolution
So: all macro-behaviour is contained in our initial set of macro-variables
U is sufficient to accurately determine overall system evolution

An iterative approach… fourth attempt

- Fourth attempt

→ Macroscopic state

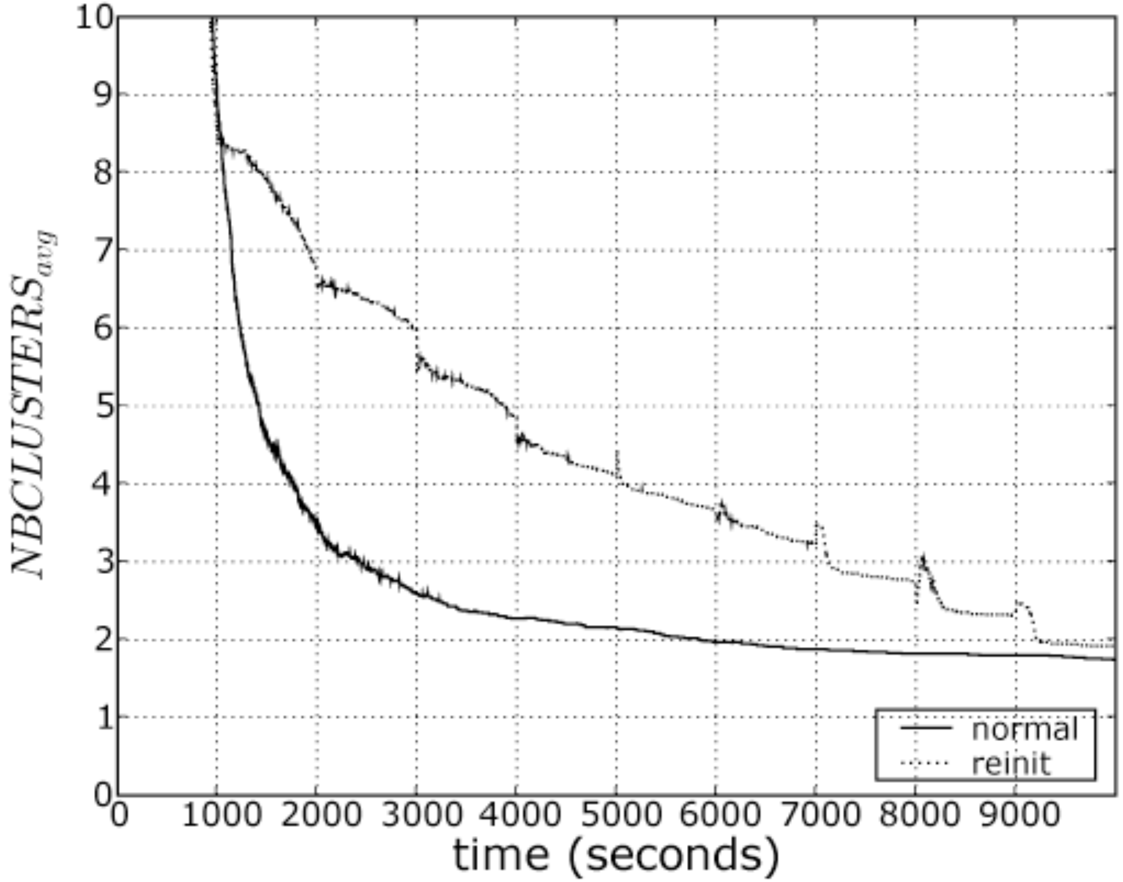
- Clusters: number
  avg distance between clusters
  ~~size~~ → avg. size

→ INIT:

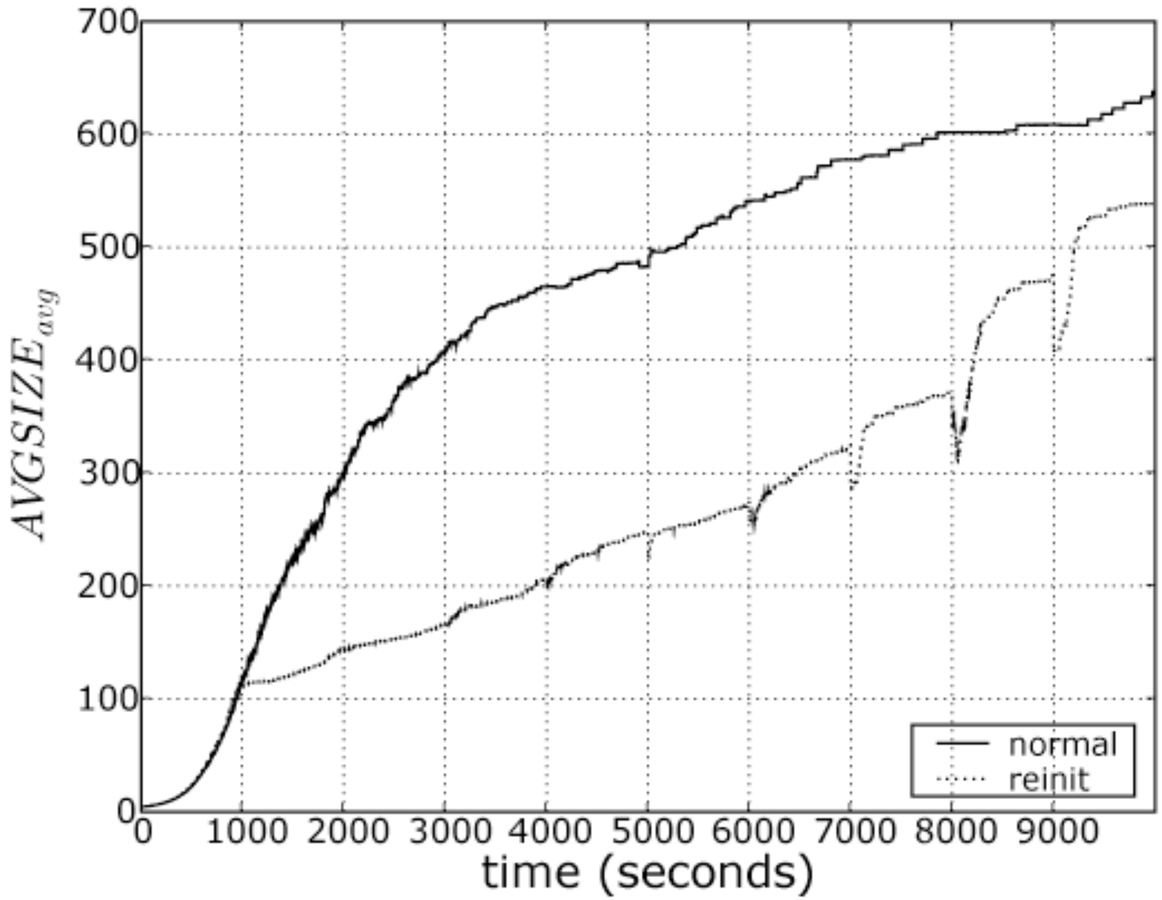
- Clusters: clusters, same avg. size, avg distance,
  exact positions of data items randomized
- Agents: randomized

Evaluation:

U is **IN**-sufficient to accurately determine overall system evolution

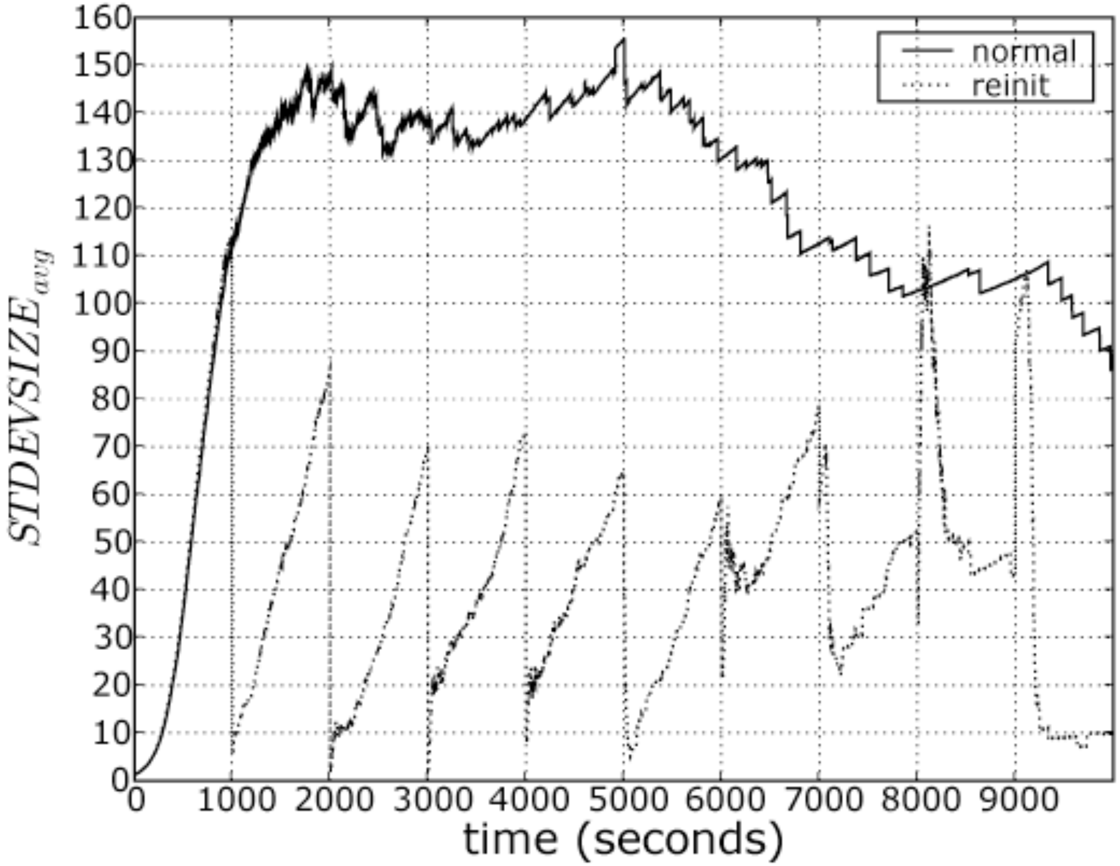


Comparison of a reference simulation (solid line)
and a simulation with reinitialisation after every 1000 time-steps → *average number of clusters*



Comparison of a reference simulation (solid line) and a simulation with reinitialisation after every 1000 time-steps → *average size of clusters*

- Standard deviation of cluster sizes, avgd



Variance is eliminated after every re-initialisation…

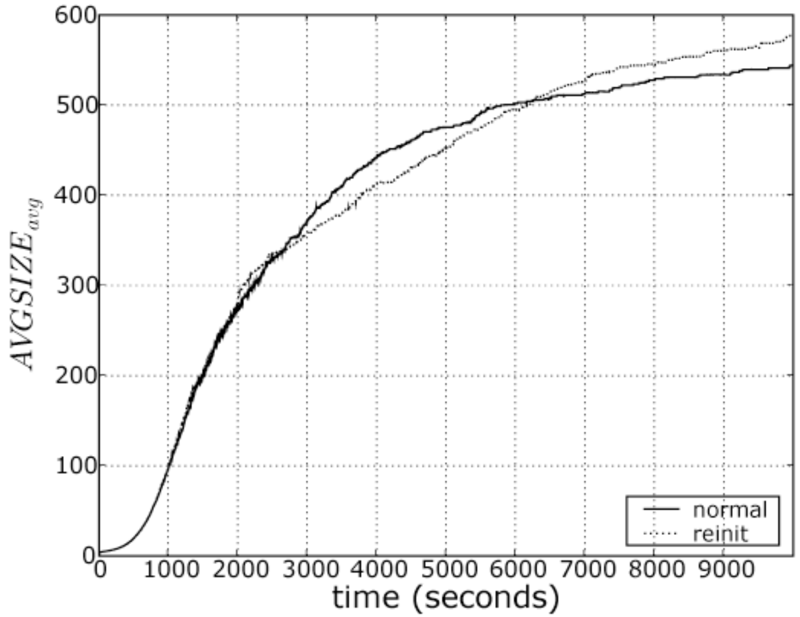
KATHOLIEKE UNIVERSITEIT LEUVEN DistriNet RESEARCH GROUP

An iterative approach… fifth attempt

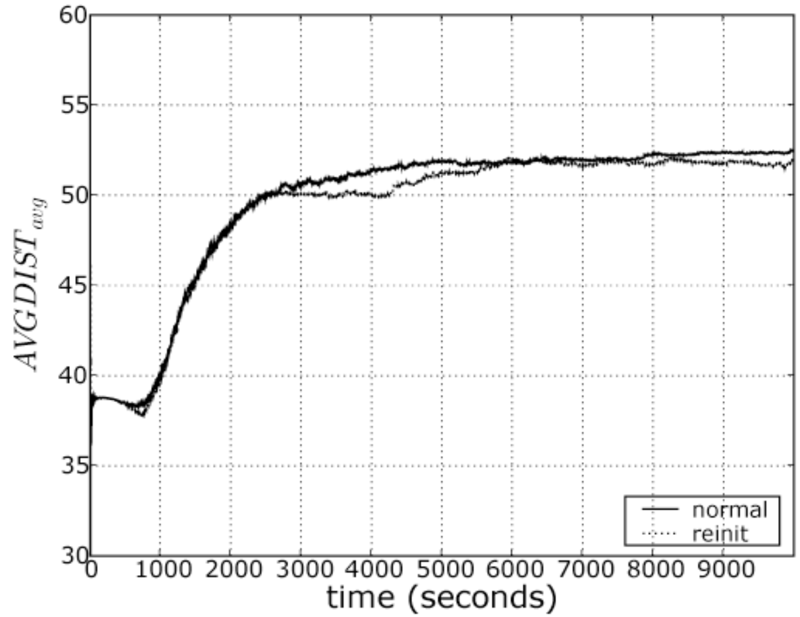
- Fifth attempt
  - → Macroscopic state
    - Clusters: number
      avg. size, avg distance between clusters
      std. deviation of cluster size
  - → INIT:
    - Clusters: clusters, same avg. size, avg distance,
      exact positions of data items randomized but
      according to std. deviation
    - Agents: randomized

Evaluation:

U is sufficient to accurately determine overall system evolution



$AVGSIZE_{avg}$
600
500
400
300
200
100
0
0 1000 2000 3000 4000 5000 6000 7000 8000 9000
time (seconds)
normal
reinit



$AVGDIST_{avg}$
60
55
50
45
40
35
30
0 1000 2000 3000 4000 5000 6000 7000 8000 9000
time (seconds)
normal
reinit

An iterative approach… fifth attempt

Gained insight:

the number of clusters decreases faster
if there is a large difference in size
and if the clusters are closer to each other

An iterative approach… n^{th} attempt

→ …

Discussion/challenges: data clustering

- With respect to the concrete case of data clustering
  - → Simple algorithm, yet not obvious to grasp the effect
  - → Gained insights in self-organising, emergent solution
    - *the presence of vacant positions is a driving force for clustering*
    - *the number of clusters decreases faster if there is a large difference in size*
    - *the number of clusters decreases faster if the clusters are closer to each other*
  - → Helps to **evaluate** the proposed 'architecture'
  - → Inspire **improvement**
    - Making agents more aware of cluster location (e.g. thrallough digital pheronomes)
    - Agents drop data item with higher probability in neighbourhood of *large* cluster
    - …

Discussion/challenges: the approach

- → Recall: the approach is an approach…
  - Which does *not tell* you what the micro-macro relation is
  - But which ONLY *helps* or *guides* you in your *study* to understand the relation

- → Critical notes:
  - It does not say WHICH macro-variables should be chosen
    - In the example:
      - One emergent… e.g. clusters (→ identify observable artefacts)
      - Rest: aggregation/abstraction through statistics (avg, variance)
  - Designing an initialisation operator becomes increasingly hard!
  - Parameters must be chosen wisely
    - Initial transient
    - Reinitialisation period
      - Short enough to be efficient (large simulations)
      - Long enough to pass transient effects

Conclusion

- Understanding SO-em behaviour is necessary but hard…
  → Evaluation / improvement of software architecture
- Iterative EFA-based approach gives some guidance
  → More research needed!

Overview

1. Delegate MAS: BDI through the environment

   Tom Holvoet, Paul Valckenaers

   LNCS - Environments for MAS 2006

   Alexander Helleboogh, Danny Weyns, Tom Holvoet, Rutger Claes
   ITS Conference 2007
   IEEE Journal on ITS (submitted)

2. "Using Equation-Free Macroscopic Analysis for Studying Self-Organising Emergent Solutions"

   Giovanni Samaey, Tom Holvoet, Tom De Wolf

   IEEE Conf. on Self-Adaptation and Self-Organisation (SASO'2008)

   Venice, Italy

Conclusion

- SASO'2009

Third International Conference on
Self-Adaptive and Self-Organizing Systems
San Francisco, California, September 14-18, 2009

The topics of interest:

* Control of emergent properties in self-* systems
* Biologically, socially, and physically inspired self-* systems
* Management and control of self-* systems
* Self-organization
* Self-adaptation
* Other self-* properties (self-management, self-monitoring, self-tuning, self-repair, self-configuration, etc.)
* Theories, frameworks and methods for self-* systems
* Robustness and dependability of self-* systems
* Approaches to engineering self-* systems
* Applications and experiences with self-* systems

Conclusion

- SASO'2009

Third International Conference on
Self-Adaptive and Self-Organizing Systems

San Francisco, California, September 14-18, 2009

Call For	research papers	(April 8)
Call For	posters	(April 23)
Call For	tutorials	(April 30)
Call For	workshops	(??)

Overview

1. Delegate MAS: BDI through the environment

   Tom Holvoet, Paul Valckenaers

   LNCS - Environments for MAS 2006

   Alexander Helleboogh, Danny Weyns, Tom Holvoet, Rutger Claes

   ITS Conference 2007

   IEEE Journal on ITS (submitted)

2. "Using Equation-Free Macroscopic Analysis for Studying Self-Organising Emergent Solutions"

   Giovanni Samaey, Tom Holvoet, Tom De Wolf

   IEEE Conf. on Self-Adaptation and Self-Organisation (SASO'2008)

   Venice, Italy

?