A Case-Based Reasoning approach for Norm adaptation

Jordi Campos[1], Maite López-Sánchez[1], Marc Esteva[2]

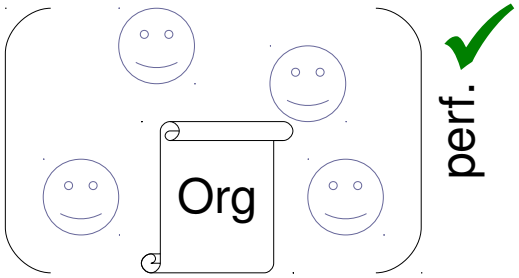
HAIS 2010

[1] Universitat de Barcelona (UB)
[2] Institut d'Intel·ligència Artificial (IIIA-CSIC)

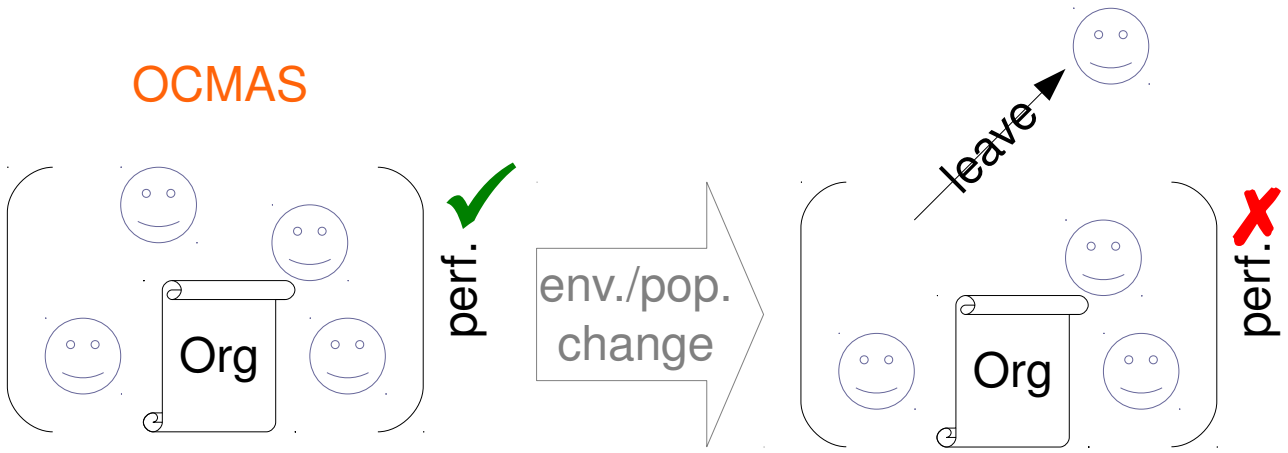
Outline

- **Motivation** (OCMAS)
- Problem characterisation
- Case Study (P2P Sharing Network)
- Our approach (2-LAMA)
- Organisational Adaptation (norm adaptation)
- Evaluation (empirical, simulation)
- Conclusions and Future Work

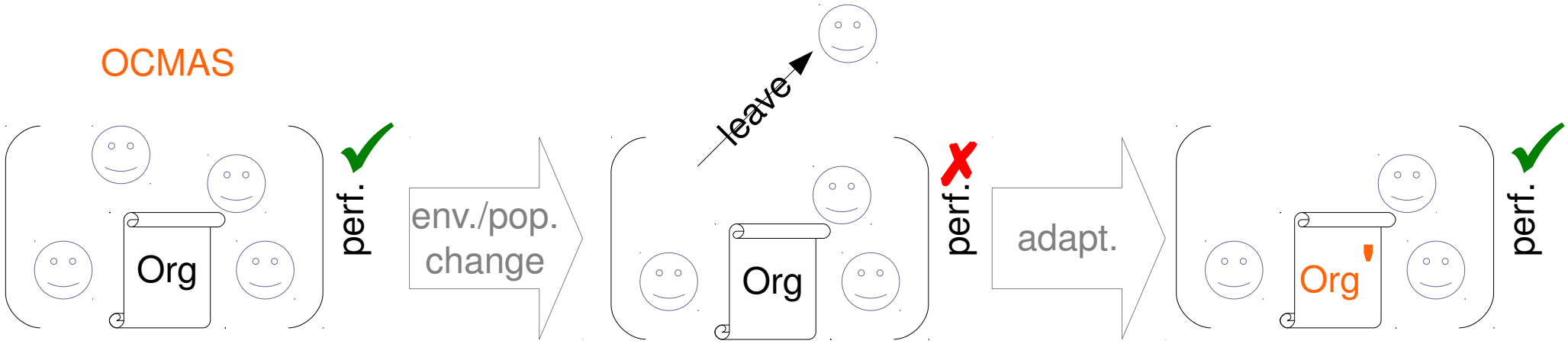
OCMAS



- Organisational-Centred MAS (**OCMAS**)
  - They have proven to be effective to regulate agents' activities (specially in open MAS & ↑dynamic dom)
  - MAS activity is regulated by an organisational struct. (**Org**) towards certain social **Goals**
    - we focus on **norms** (an Org component)



- environmental / population **changes**
  - ↓ Org. effectiveness



- environmental / population changes
  - ↓ Org. effectiveness → Org **adaptation**
    - we focus on **norm adaptation**, but we also have social structure adaptation.

Outline

- Motivation (OCMAS)
- **Problem characterisation**
- Case Study (P2P Sharing Network)
- Our approach (2-LAMA)
- Organisational Adaptation (norm adaptation)
- Evaluation (empirical, simulation)
- Conclusions and Future Work

Problem characterisation

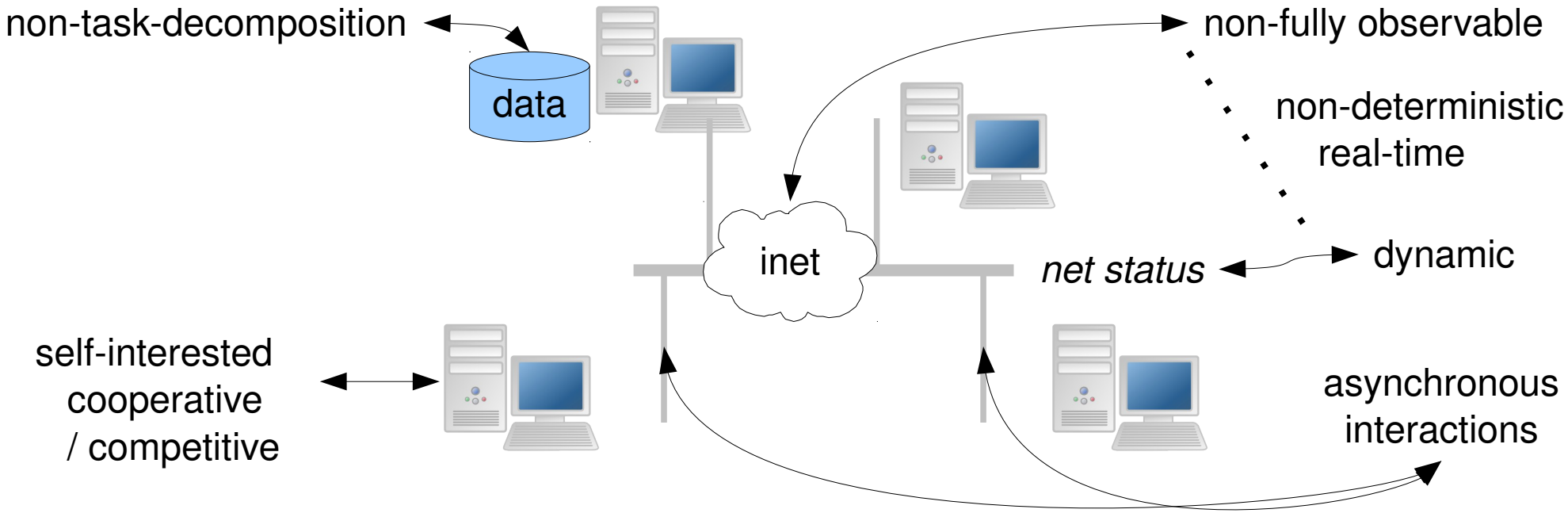
- **Environment**: [Russell&Norvig 95] [Wooldridge02]
  - non-task-decomposition oriented
    - → we are interested in **using norms to influence in agent behaviour** instead of assigning tasks
  - non-fully observable → non-deterministic
  - dynamic, real-time, run-time adaptation
- **Agent pop.**: self-interest, coop./comp., open

↳ there **exist** real **problems** with such **features**
e.g. a traffic scenario or a P2P sharing network

Outline

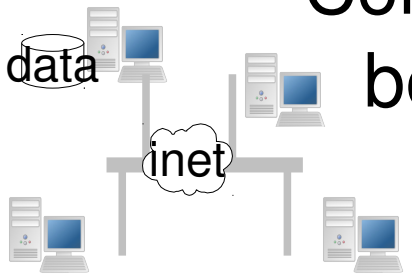
- Motivation (OCMAS)
- Problem characterisation
- **Case Study** (P2P Sharing Network)
- Our approach (2-LAMA)
- Organisational Adaptation (norm adaptation)
- Evaluation (empirical, simulation)
- Conclusions and Future Work

- a simplified P2P **Sharing network**
  - To share 1 piece of data among all connected computers (**peers**)
  - Goal: consuming the **minimum time**



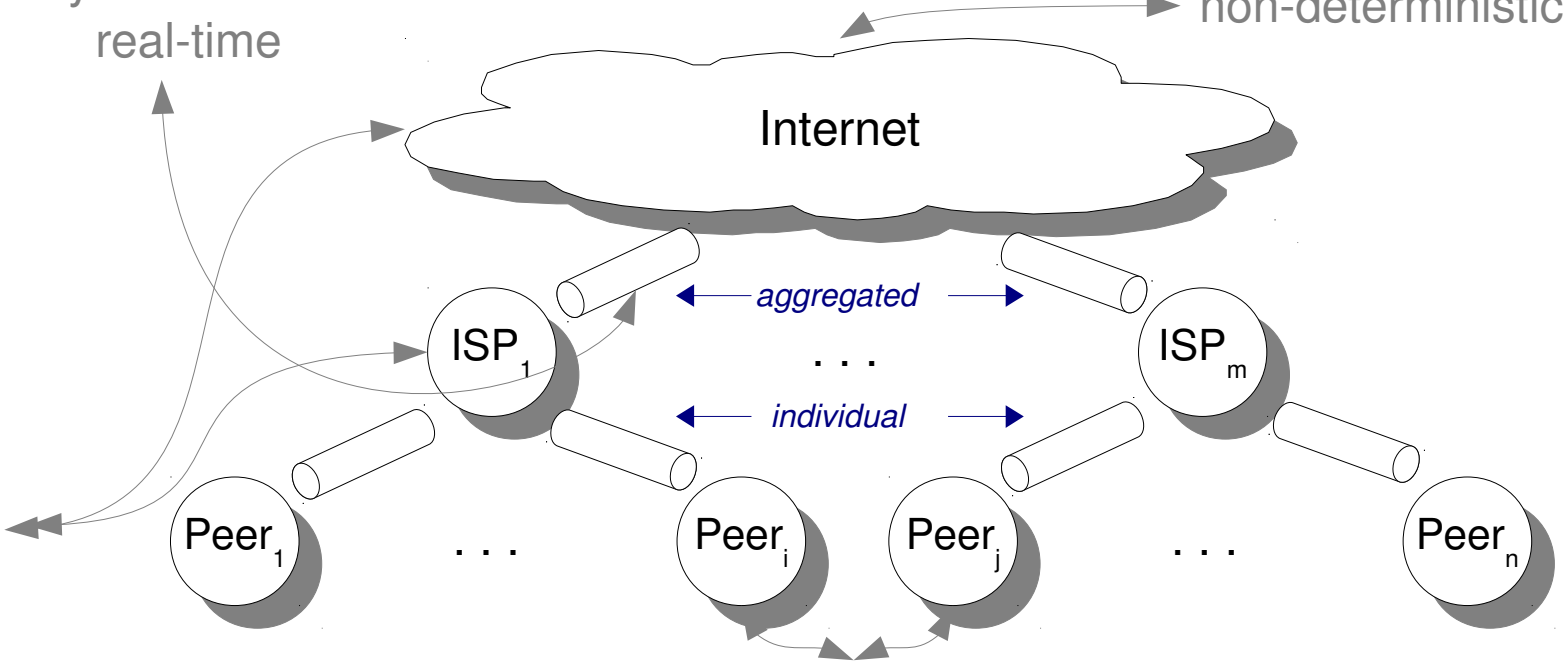
- **Network abstraction**

  - Communication channels are shared and may be saturated



dynamic
real-time

non-deterministic

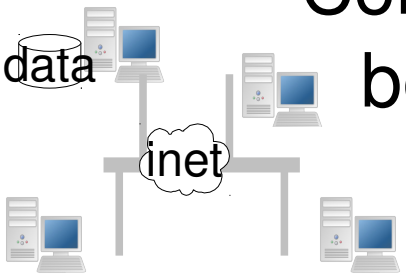


non-fully observable
- inet inaccessible
- privacy issues among ISPs

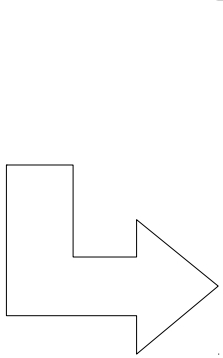
asynchronous interactions

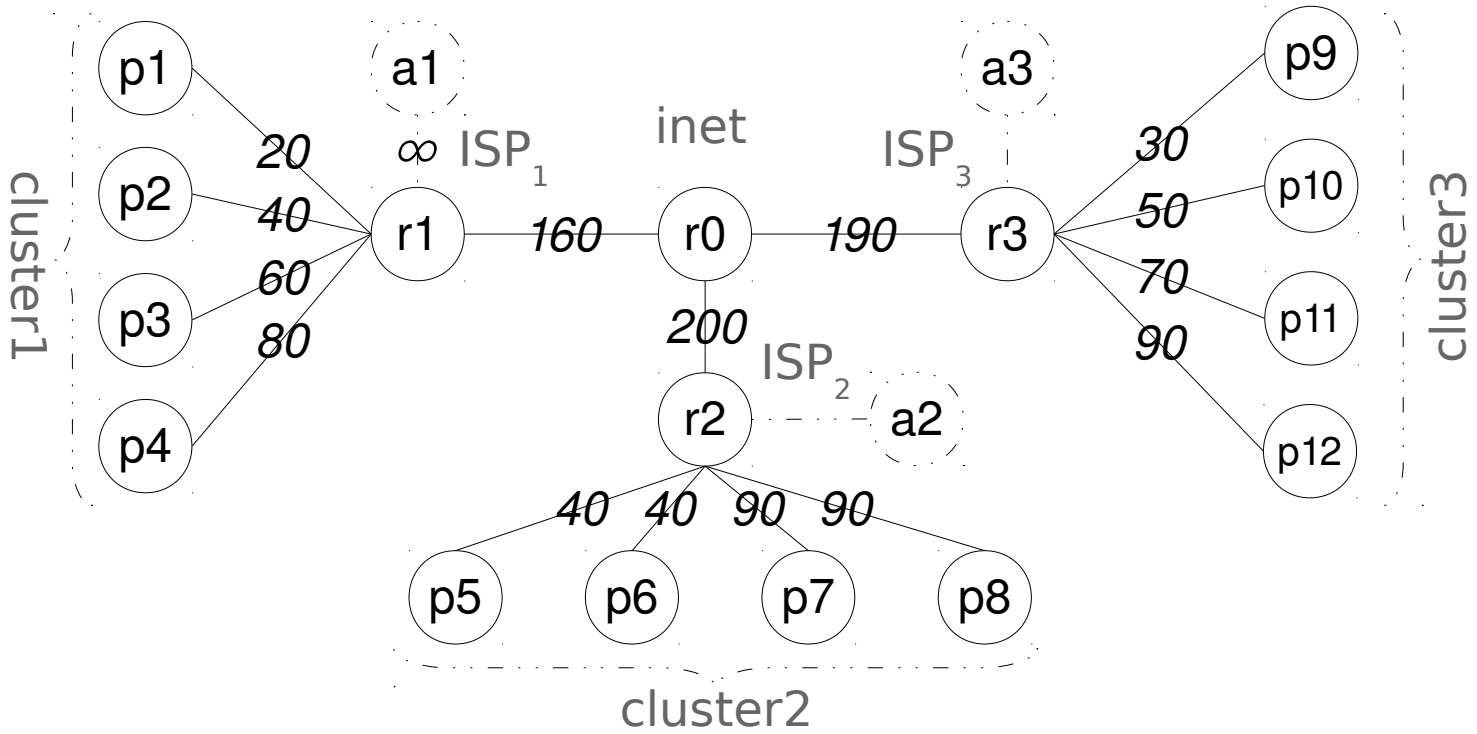
- **Network abstraction**

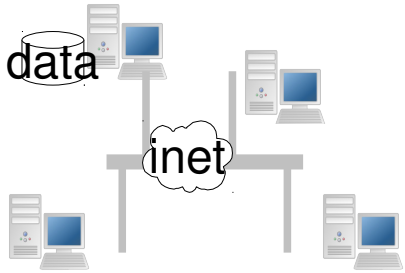
  - Communication channels are shared and may be saturated → Packet switching network

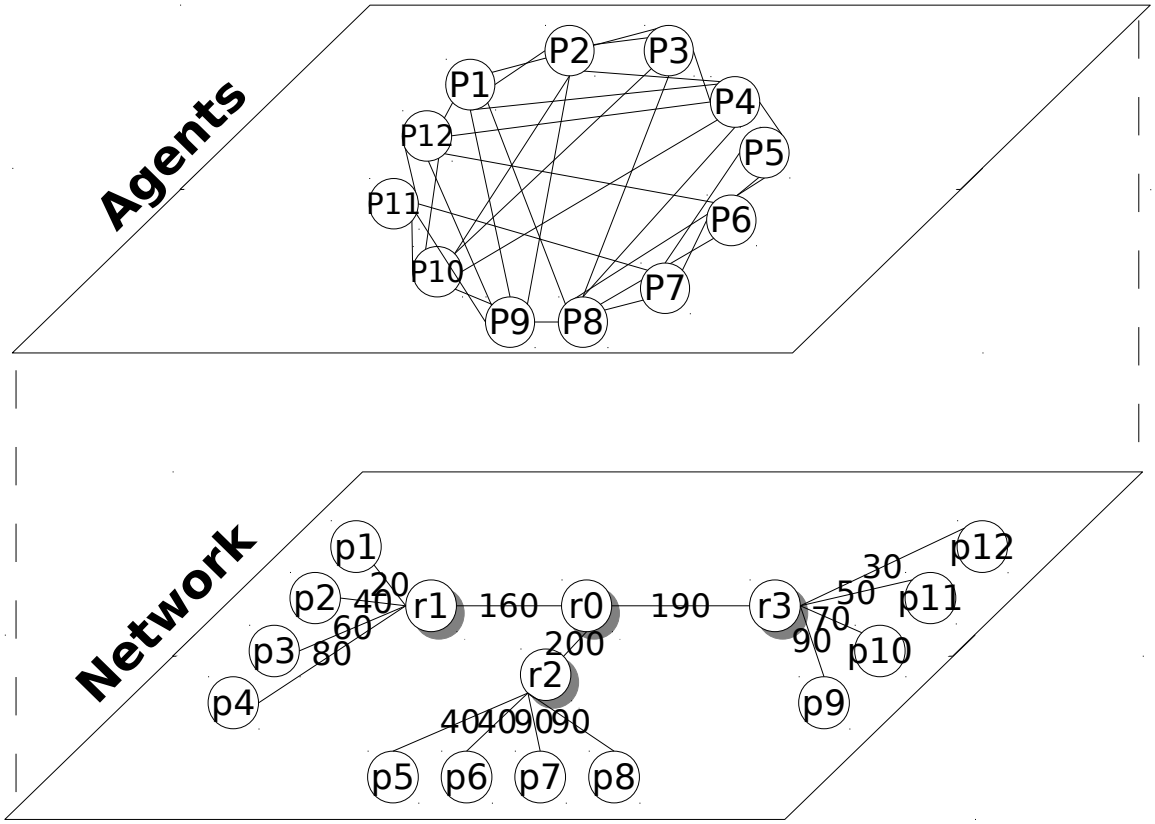


  - Msg_{txtime} = f(msg.length, #links, links.usage)









- **OCMAS** view:
  - Comput. = **Agents**
  - Net = **Environment**
  - Protocols, Social struc., Restrictions = **Organisation**

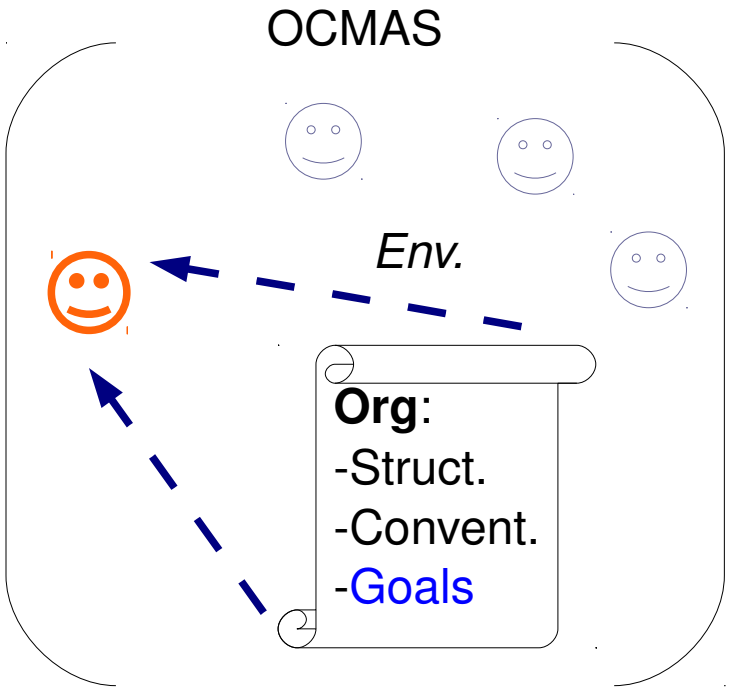
↪ **org. adaptation** to env./pop. changes may improve perf.

Outline

- Motivation (OCMAS)
- Problem characterisation
- Case Study (P2P Sharing Network)
- **Our approach** (2-LAMA)
- Organisational Adaptation (norm adaptation)
- Evaluation (empirical, simulation)
- Conclusions and Future Work

Agent **features** required to deal with **organisational issues**:

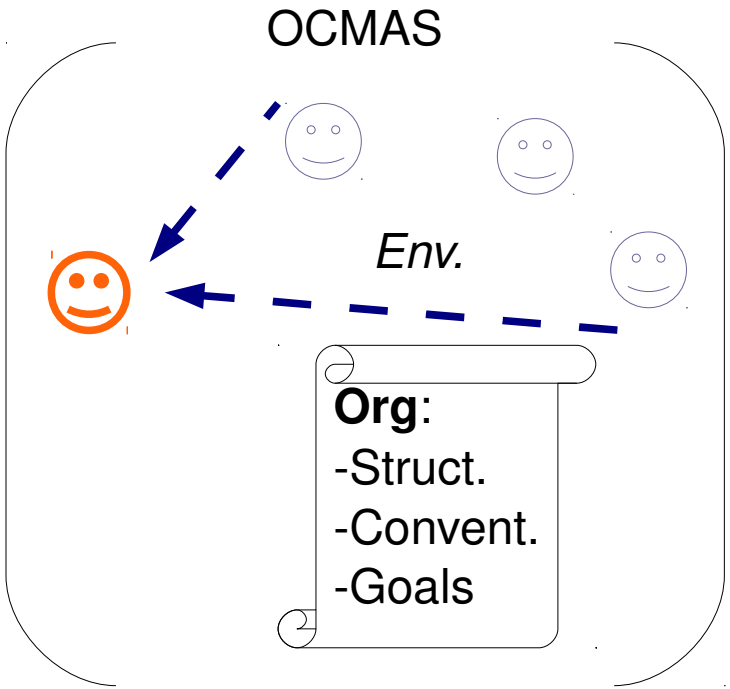
- **to Reason**
  - at a higher level of abstr.
  - considering system goals



Our approach: features

Agent **features** required to deal with **organisational issues**:

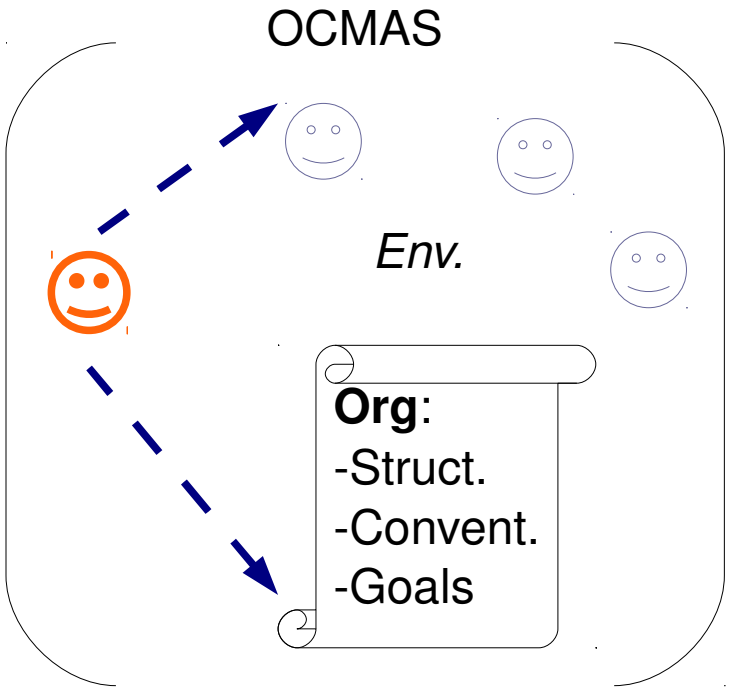
- to Reason
  - at a higher level of abstr.
  - considering system goals
- to **Perceive** certain info.



Agent **features** required to deal with **organisational issues**:

- to Reason
  - at a higher level of abstr.
  - considering system goals
- to Perceive certain info.
- **Trusted** by others

  (or ~authority)



Approach: an abstract architecture w/ 2 levels

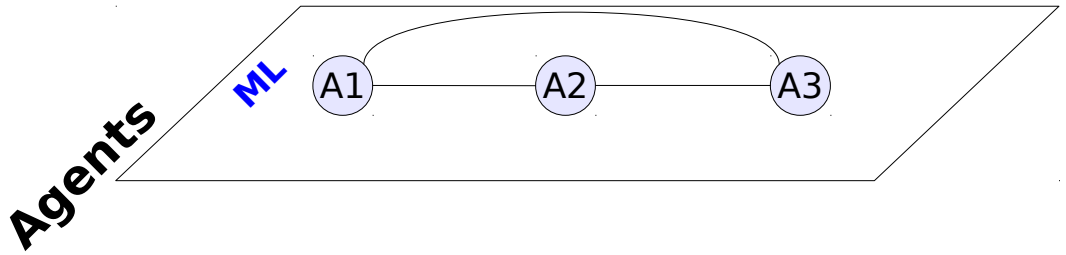
- Meta-Level (**ML**)
  - *staff* agents

    organised

    to deal with organisational issues

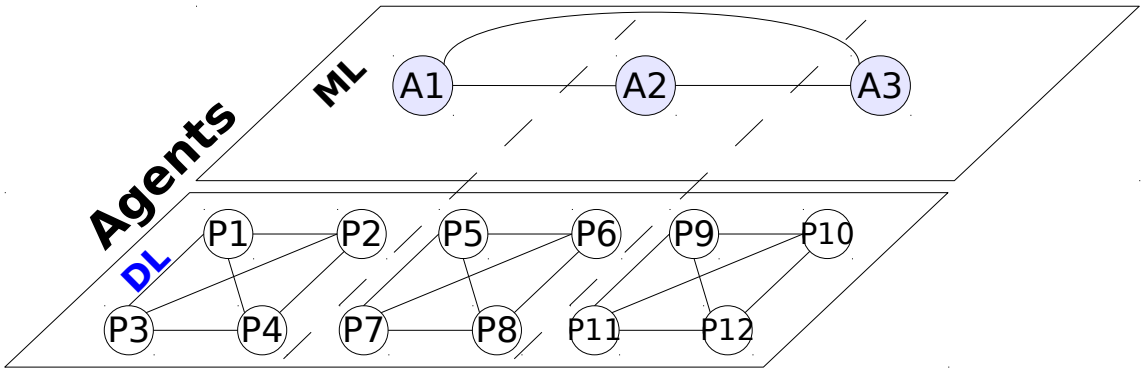
    = **assistants**
  - they present previous mentioned features

    (reasoning high level, considering social goals, accessing certain info, trusted by others)



Approach: an abstract architecture w/ 2 levels

- Meta-Level (ML)
  - assists DL
- Domain-Level (**DL**)
  - Agents organised to perform domain's activity

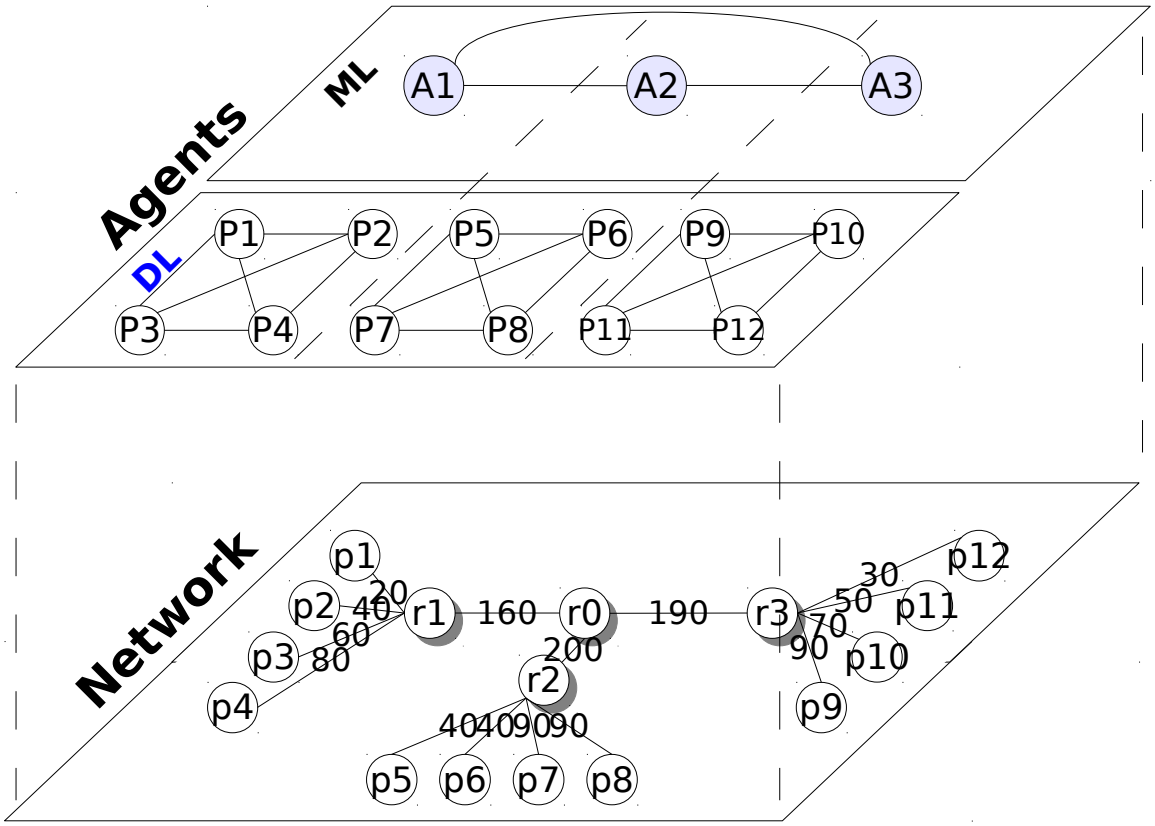


Approach: an abstract architecture w/ 2 levels

- Meta-Level (ML)
  - assists DL
- Domain-Level (DL)
  - domain's activity

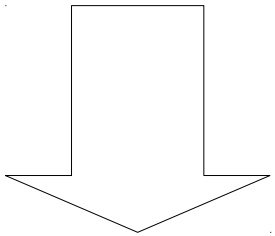
    ↪ **e.g. in P2P**:

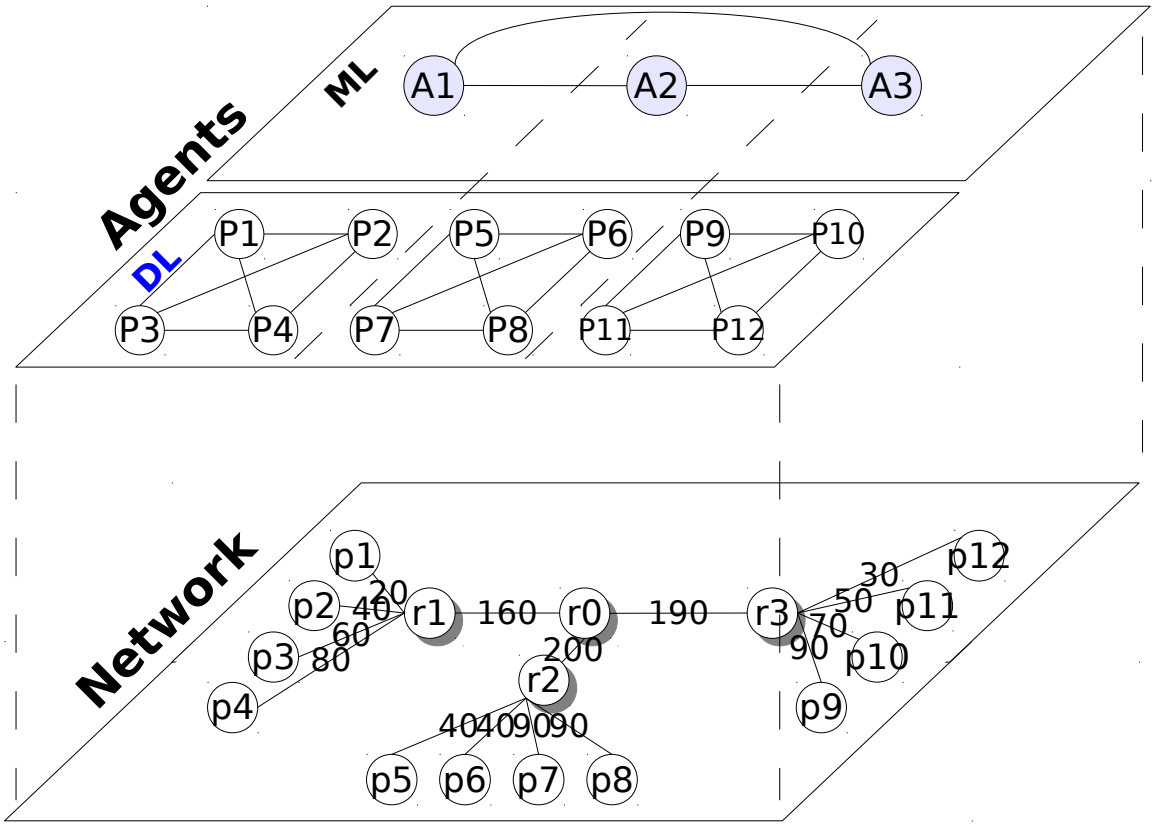
    peers that share data over a *network*



Approach: an abstract architecture w/ 2 levels

- Meta-Level (ML)
  - assists DL
- Domain-Level (DL)
  - domain's activity

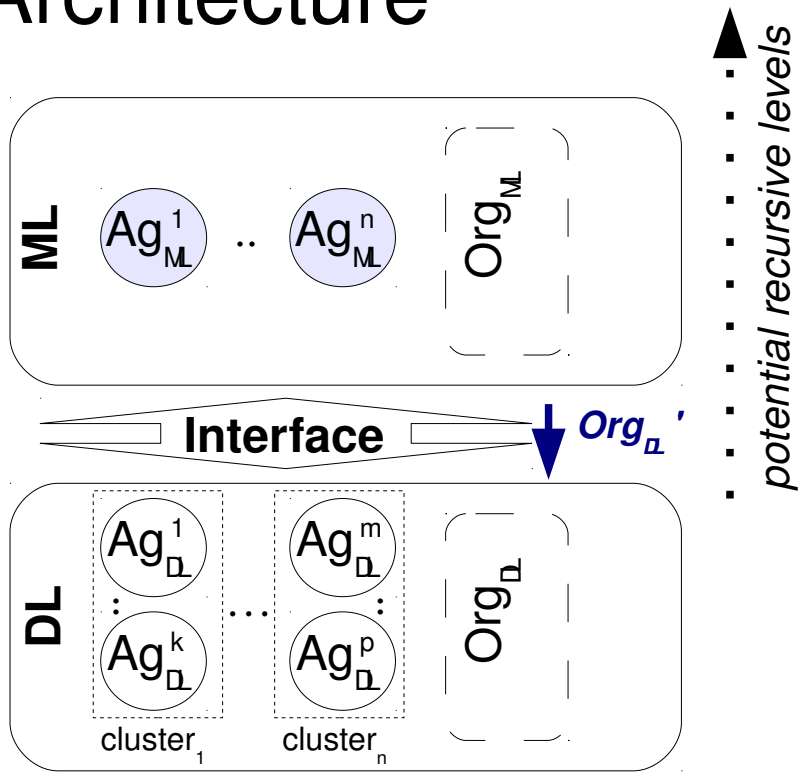




2-LAMA: Two Level Assisted MAS Architecture

Model: Two Level Assisted MAS Architecture

$$
\begin{aligned}
2LAMA = &\quad ML \times DL \times Int \\
ML &= Ag_{ML} \times Org_{ML} \\
DL &= Ag_{DL} \times Org_{DL}
\end{aligned}
$$



- ML provides **assistance serv.** to DL (e.g. Org. Adaptation)

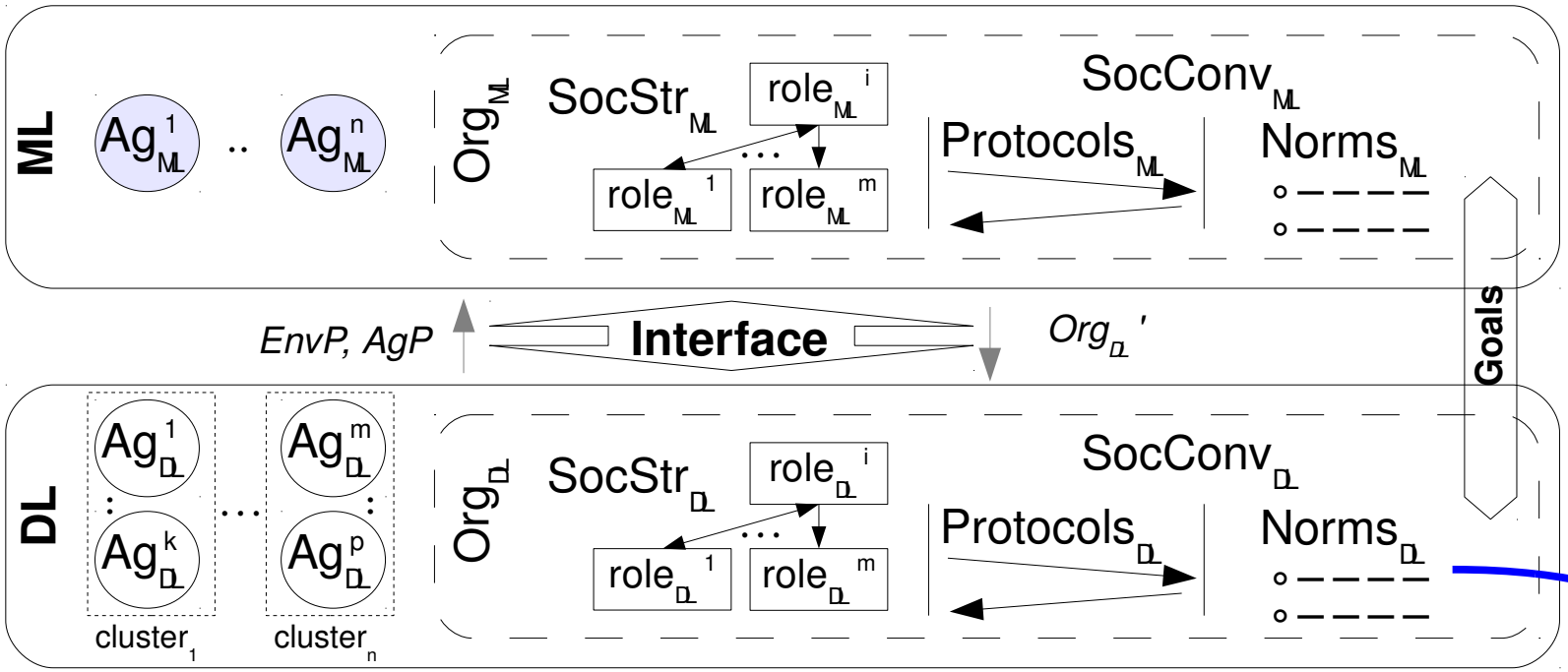
  ↪ Divison of labour

  - ML/DL are solving totally different problems
  - ↑trust to reason about social goals (e.g. like politicians, who cannot be involved in activities they regulate)

Outline

- Motivation (OCMAS)
- Problem characterisation
- Case Study (P2P Sharing Network)
- Our approach (2-LAMA)
- **Organisational Adaptation** (norm adaptation)
- Evaluation (empirical, simulation)
- Conclusions and Future Work

Org. Adaptation: norms

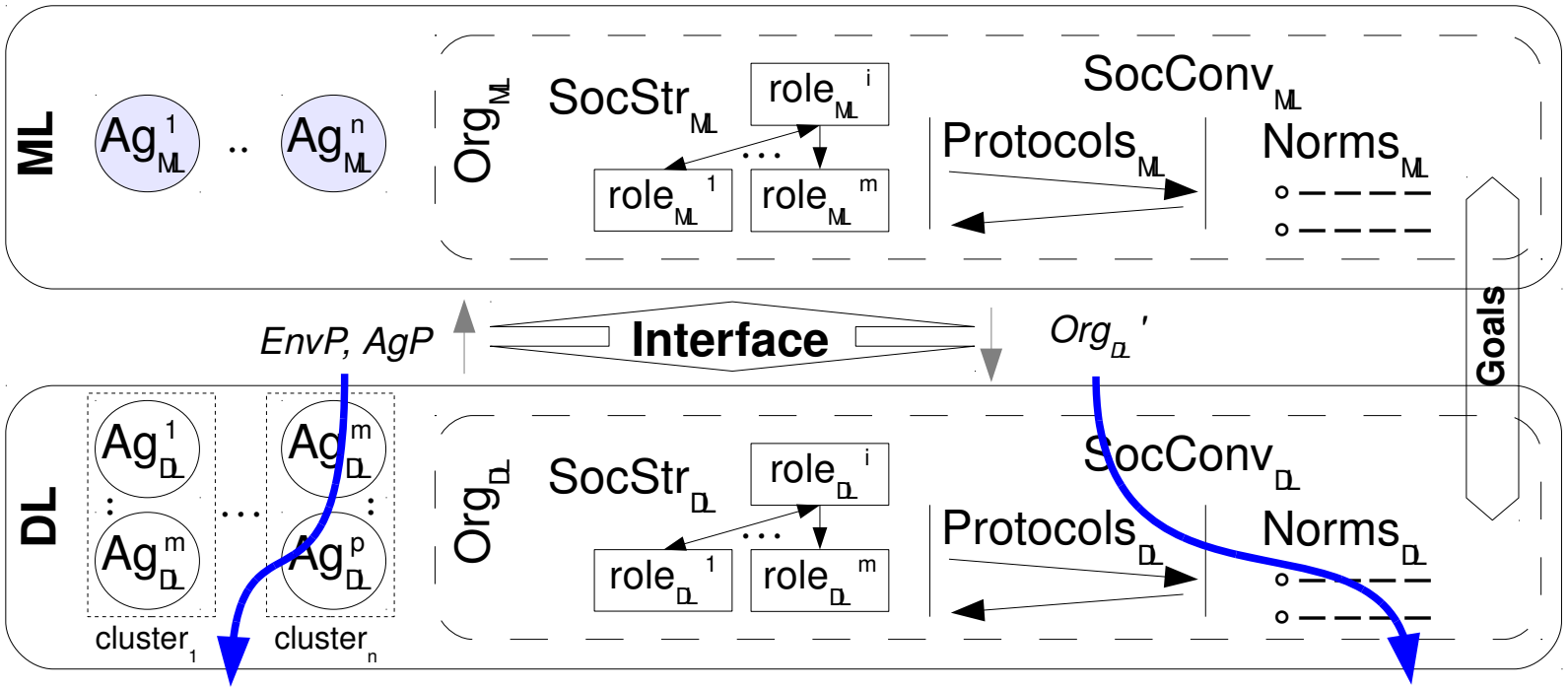


- **DL**: org = (socstr, {prot, norms}, goals)

in P2P

- $norm_{Friends}$: "a peer cannot send data to >maxFR simult."
- $norm_{BW}$: "a peer cannot use >maxBW bandwidth%"

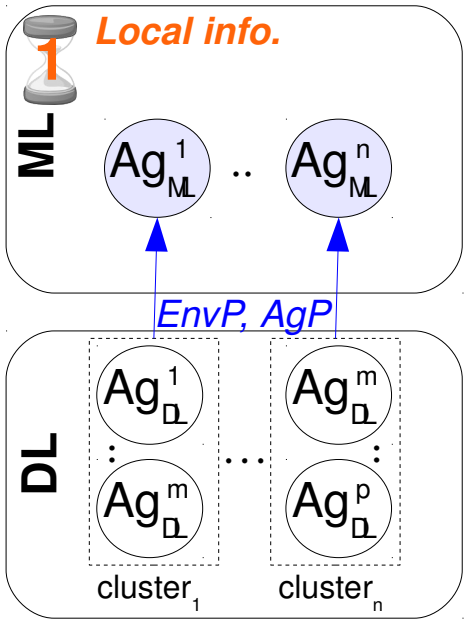
Org. Adaptation: in/out



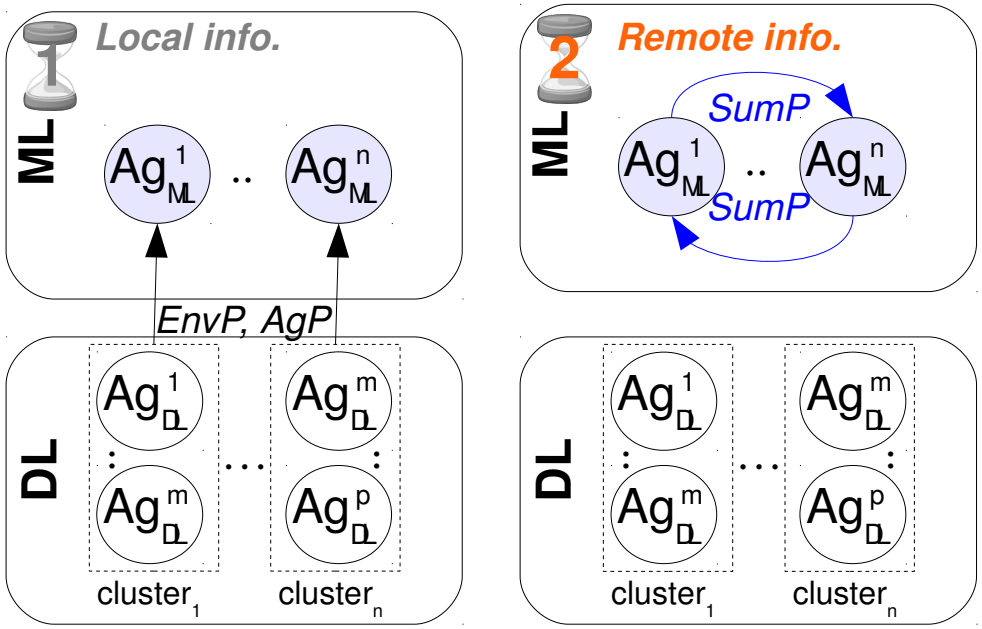
- **ML**: α^N: EnvP x AgP x Norm x Goals → Norm
  - $\alpha^N = \beta^N (\{\alpha_1^N .. \alpha_n^N\})$
  - Assist: α_i^N: $EnvP_i$ x AgP_i x$(SumP_j)^{n-1}$ x N x G → N

In current implementation: β^N = ***voting*** & α_i^N = **Heuristic** / ***CBR***

Org. Adaptation: one step

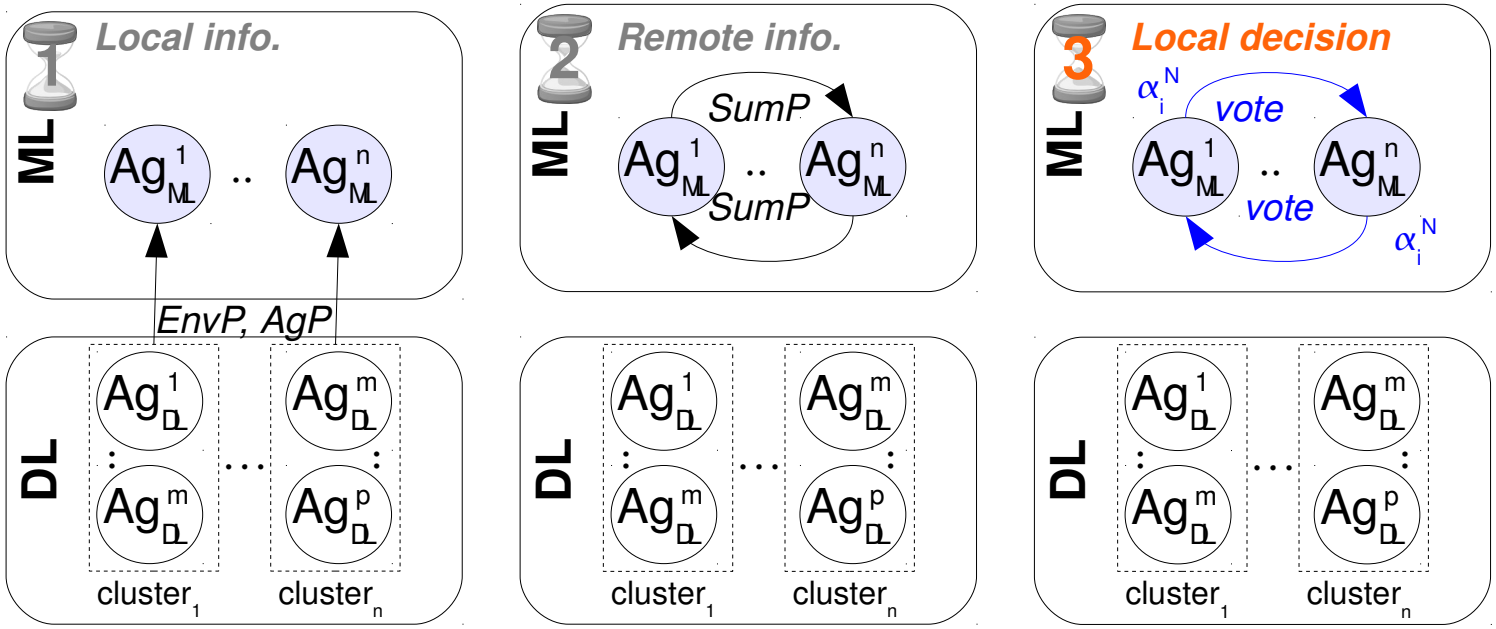


- **EnvP**: nominal&real BW for each individual link
- **AgP**: the % of data possesed by each peer



- **SumP**: statistic aggregation of EnvP and AgP
  - **SrvBW**: nominal BW of peers that are serving
  - **RcvBW**: nominal BW of peers that are receiving
  - **RcvEffBW**: real receiving BW
  - **Wait**: #incomplete_peers that are not receiving

Org. Adaptation: one step



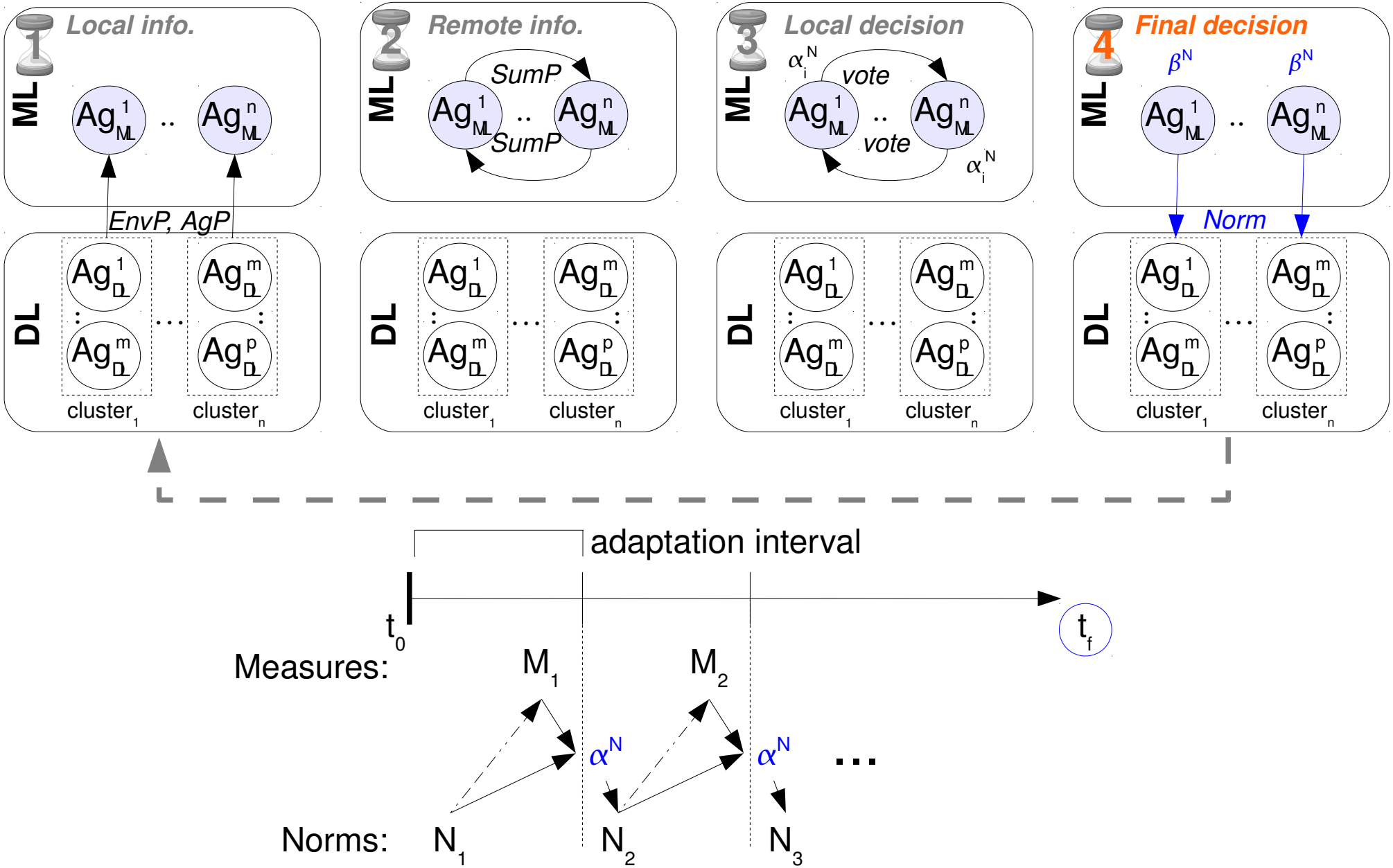
- $\alpha_i^N: EnvP_i \times AgP_i \times (SumP_j)^{n+1} \times Norm \times Goals \rightarrow Norm$

Learning technique: CBR

- use previous **experiences to reason** about current situation
- if $\nexists$ confident previous experience → use *Heuristic* to suggest new norms

- to align the serving BW capacity with the receiving one
- if the effective received BW is smaller than serving one, there is net saturation

Org. Adaptation: one step

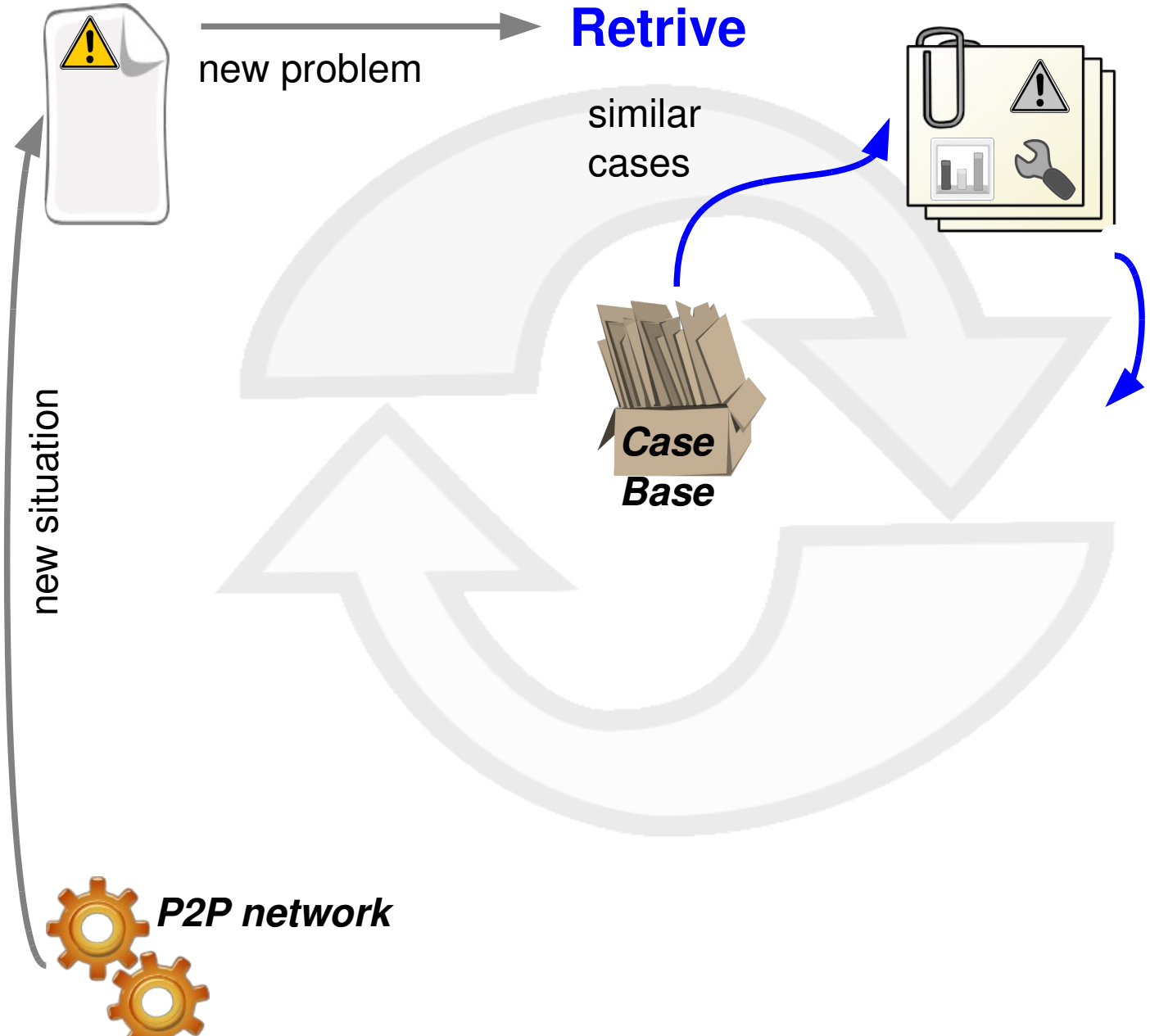


Case

- **Problem** (attributes/features): discretised|*continuous*
  - srvCapacity = SrvBw vs. RcvBW (<<,<,=,>,>>)
  - netSat = RcvBW vs. RcvEffBW (<<,<,=,>,>>)
  - waiting = wait (↓,→,↑)
  - maxShareRatio = maxFR (↓,→,↑)
  - bandwidthUsage = maxBW (↓,→,↑)
  - *executionPhase = DOC (↓,→,↑)*
- **Solution**:
  - vFR: vote about maxFR (↑,=,↓,⊘)
  - vBW: vote about maxBW (↑,=,↓,⊘)
- ***Evaluation**:*
  - *goodness = f(DOCbefore, DOCafter, final_time)*

(under construction)

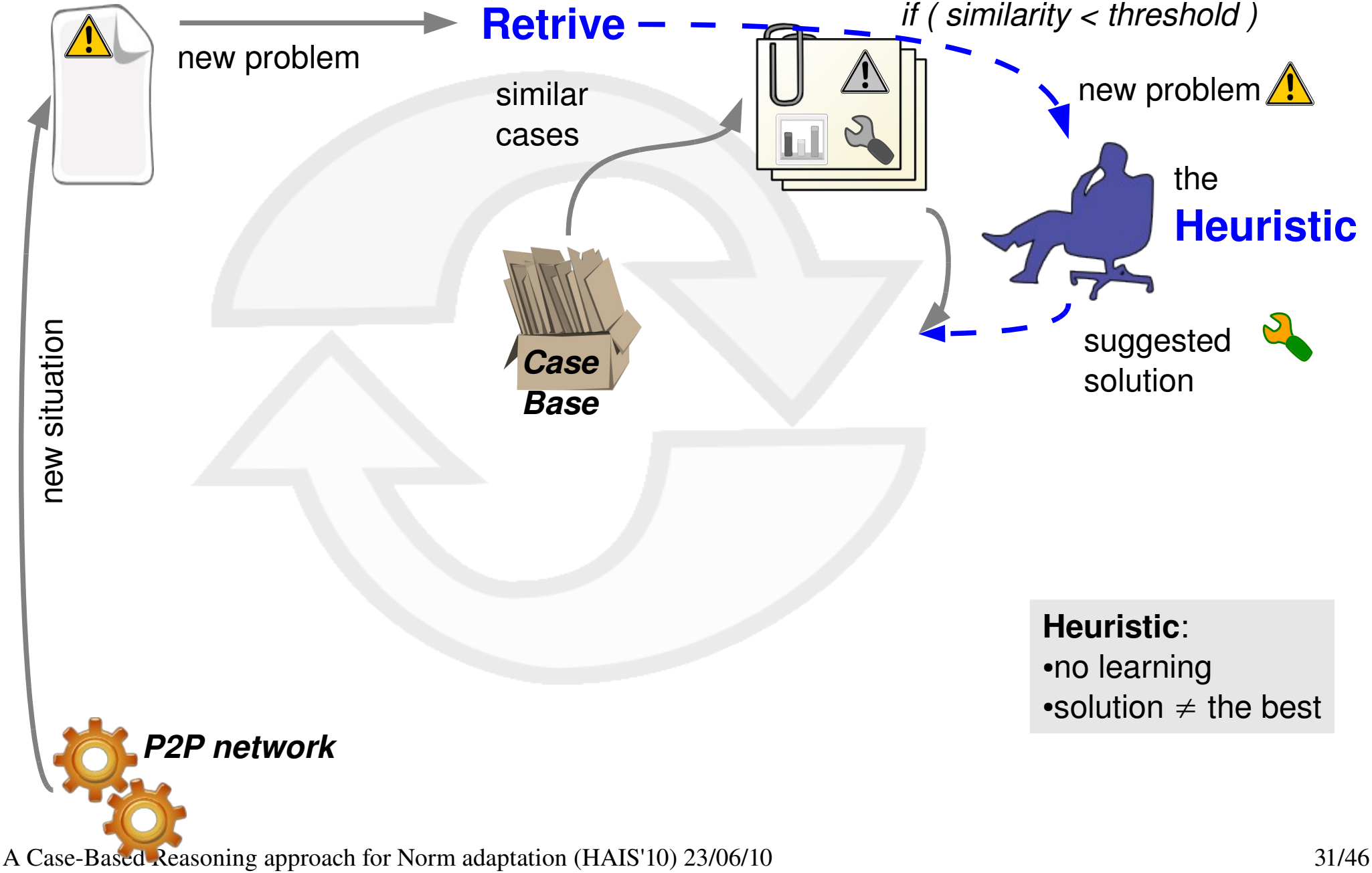
CBR: retrieve



Similarity:
=weighted difference among attributes
- discrete labels are converted into integers

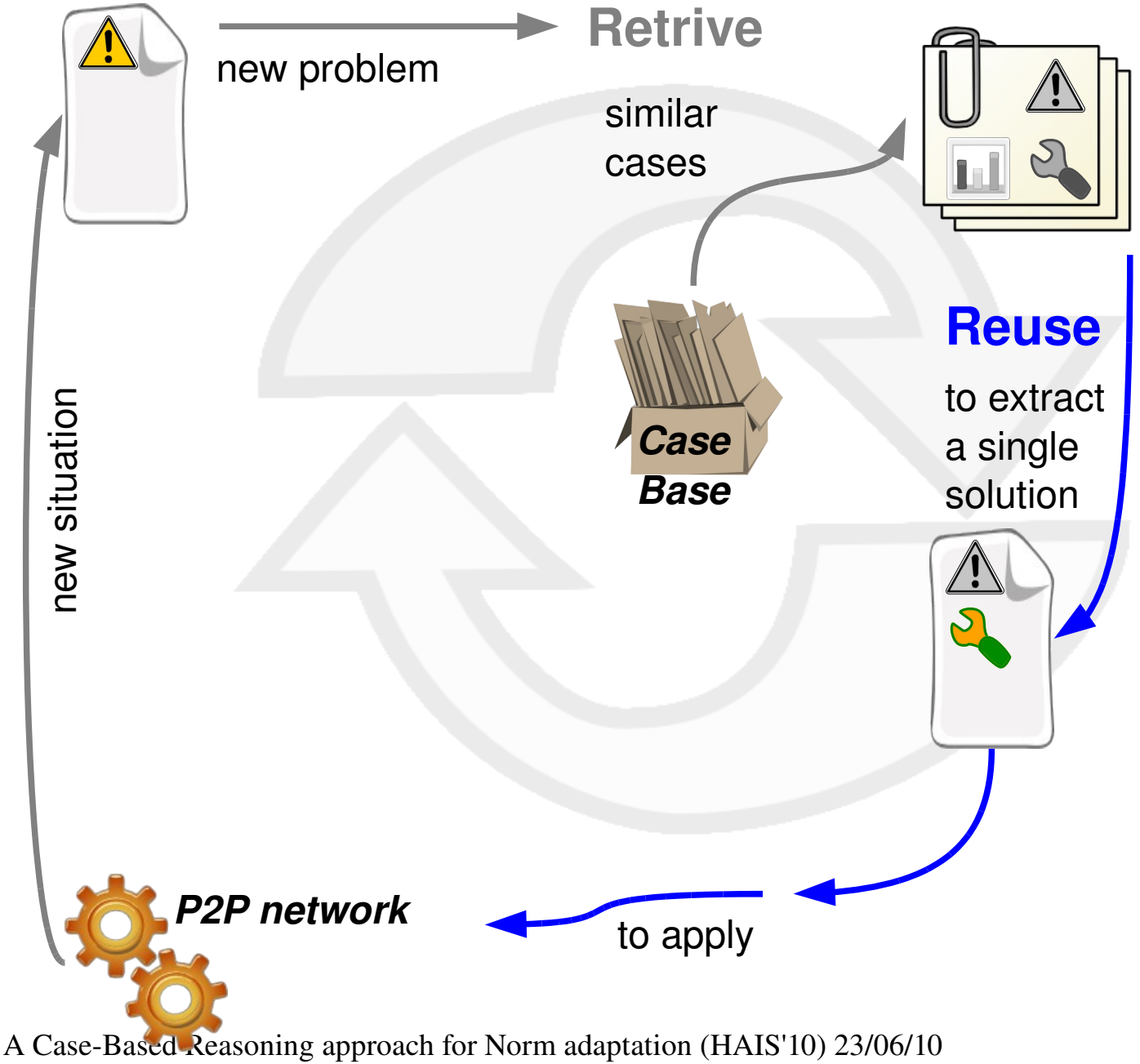
Confidence:
Problem's similarity threshold

CBR: retrieve



Heuristic:
- no learning
- solution ≠ the best

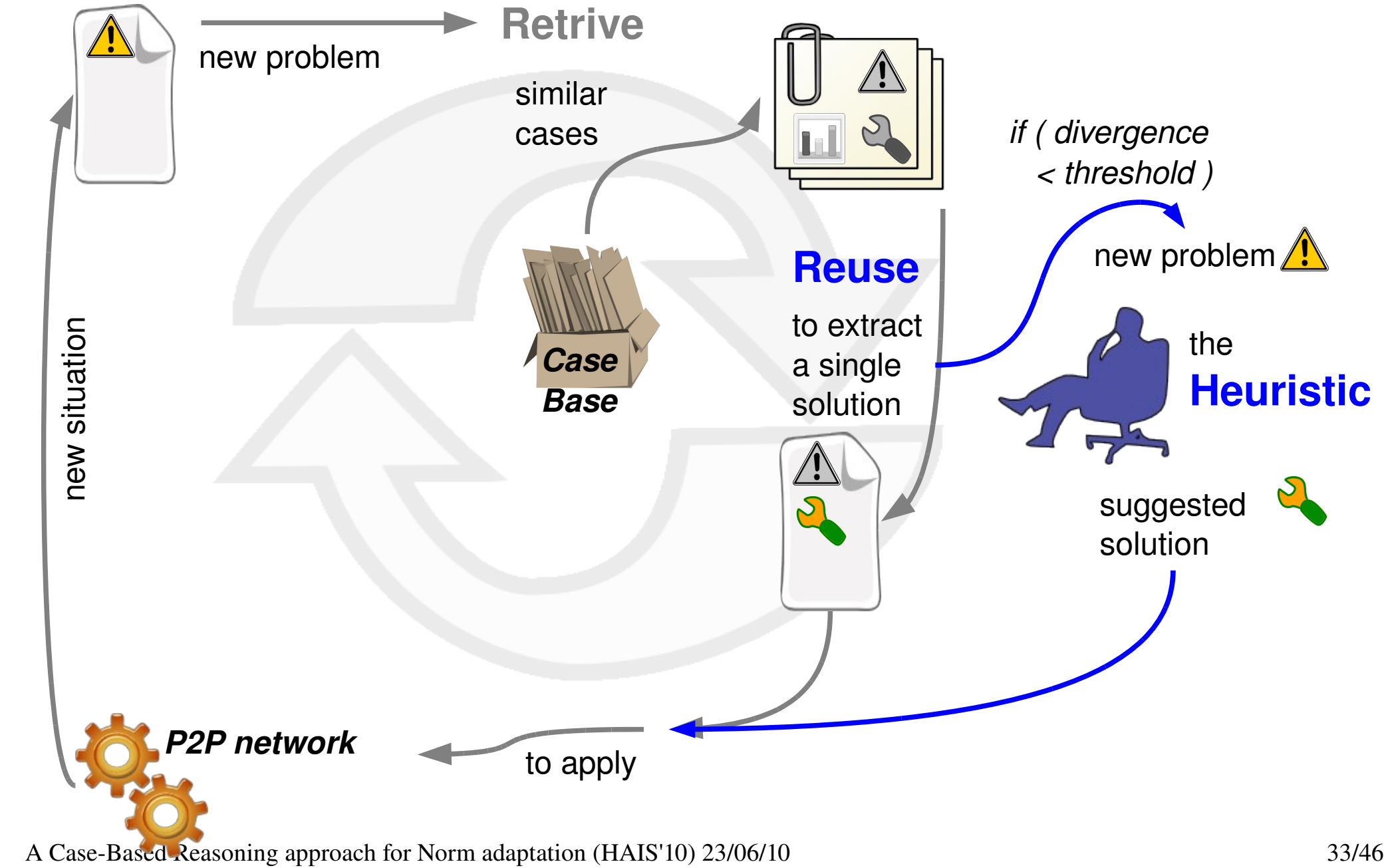
CBR: reuse



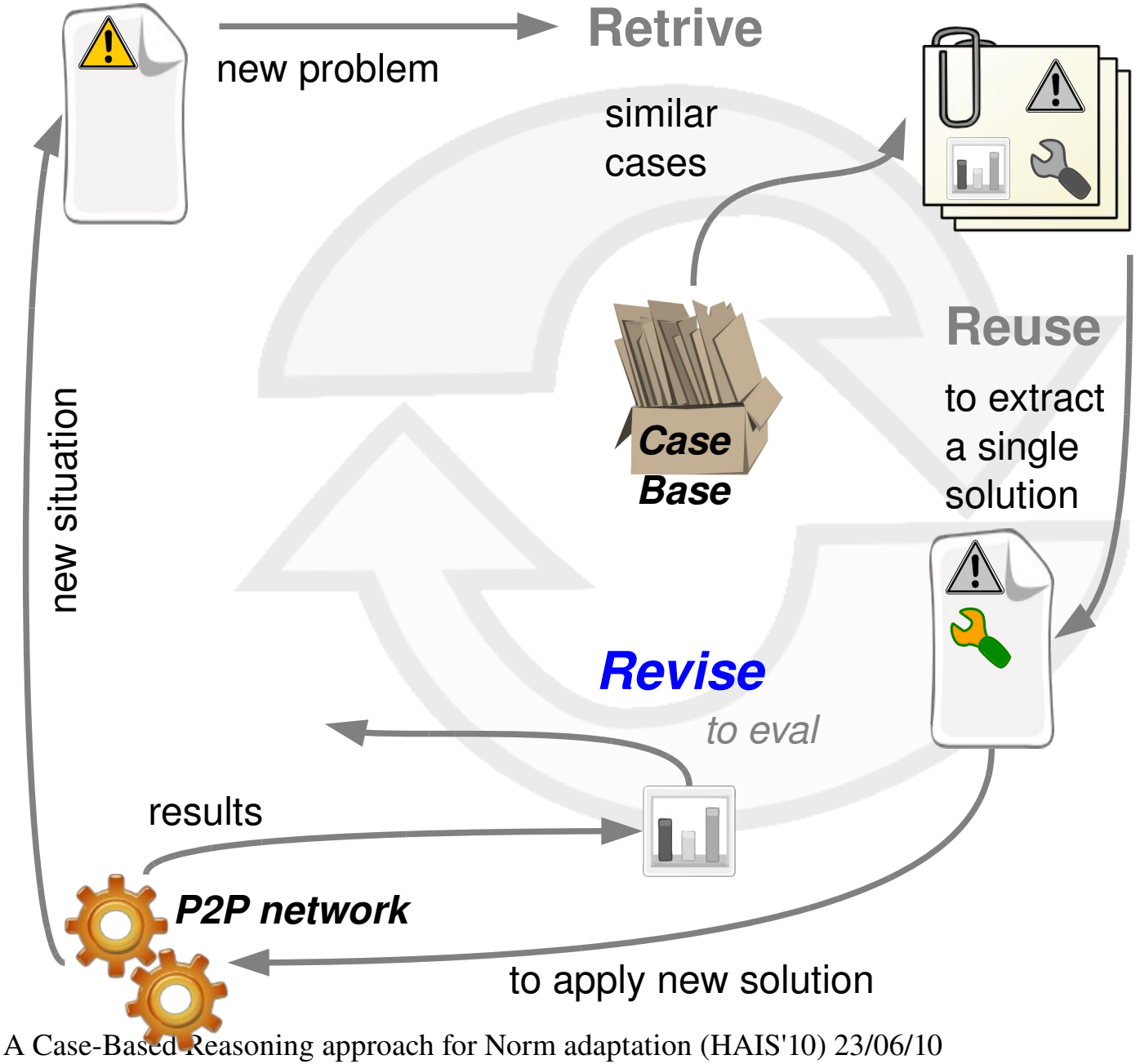
Adapt():
=voting among different solutions
- tie → extrem opt. win
- tie(extreme opt.) → change nothing

Confidence:
Solution's Divergence is is the difference among vFR converted into integers

CBR: reuse


Retrive
new problem
similar
cases
Case
Base
Reuse
to extract
a single
solution
if (divergence
< threshold)
new problem
the
Heuristic
suggested
solution
new situation
P2P network
to apply

CBR: revise

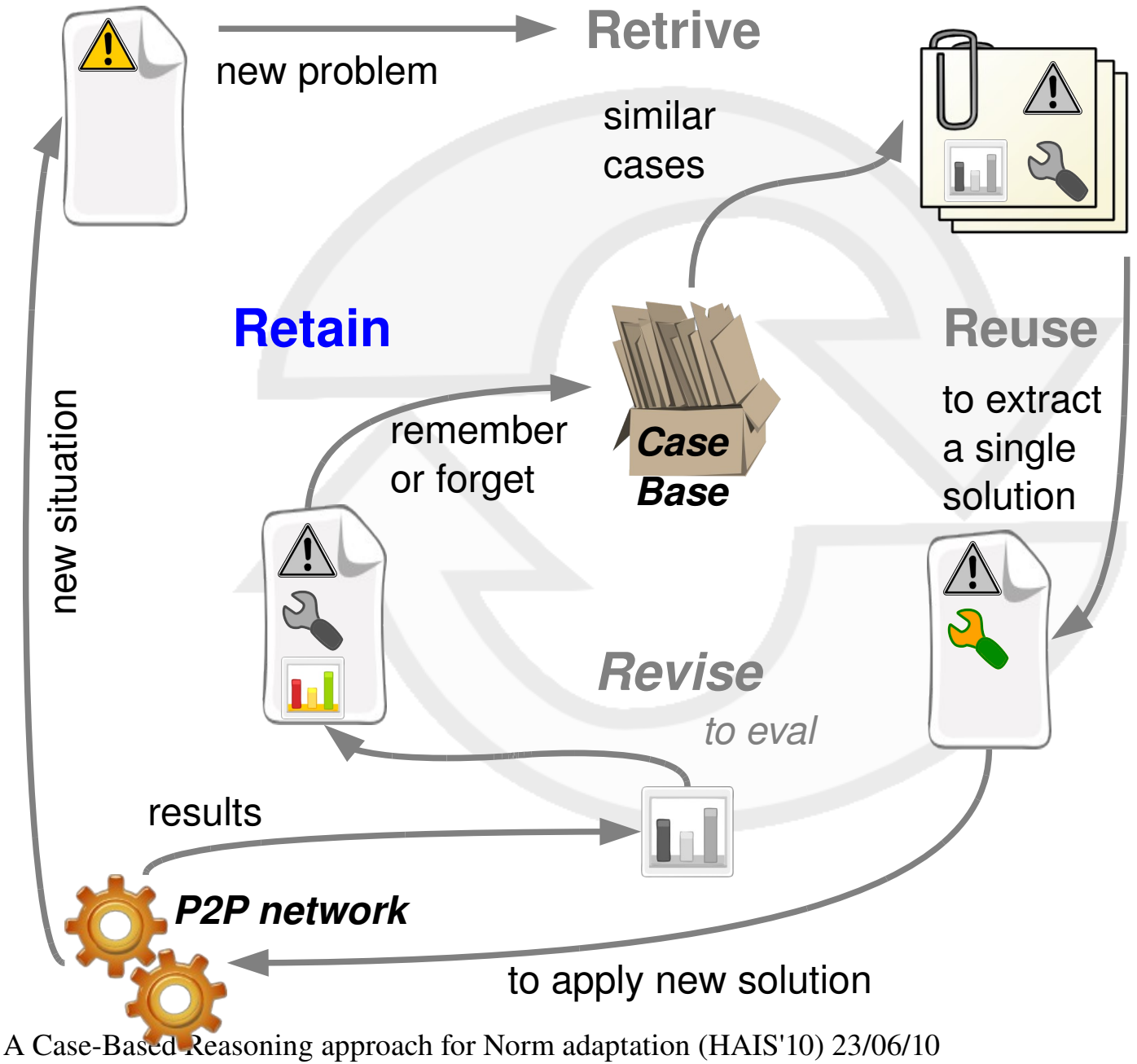


We are currently working on Revise:

Evaluation is based on:

- incremental degree of completeness

- final sharing time



Paper:
Save all cases suggested by Heuristics (it was used when there was a low confidence in current Case Base)
= Learning from others

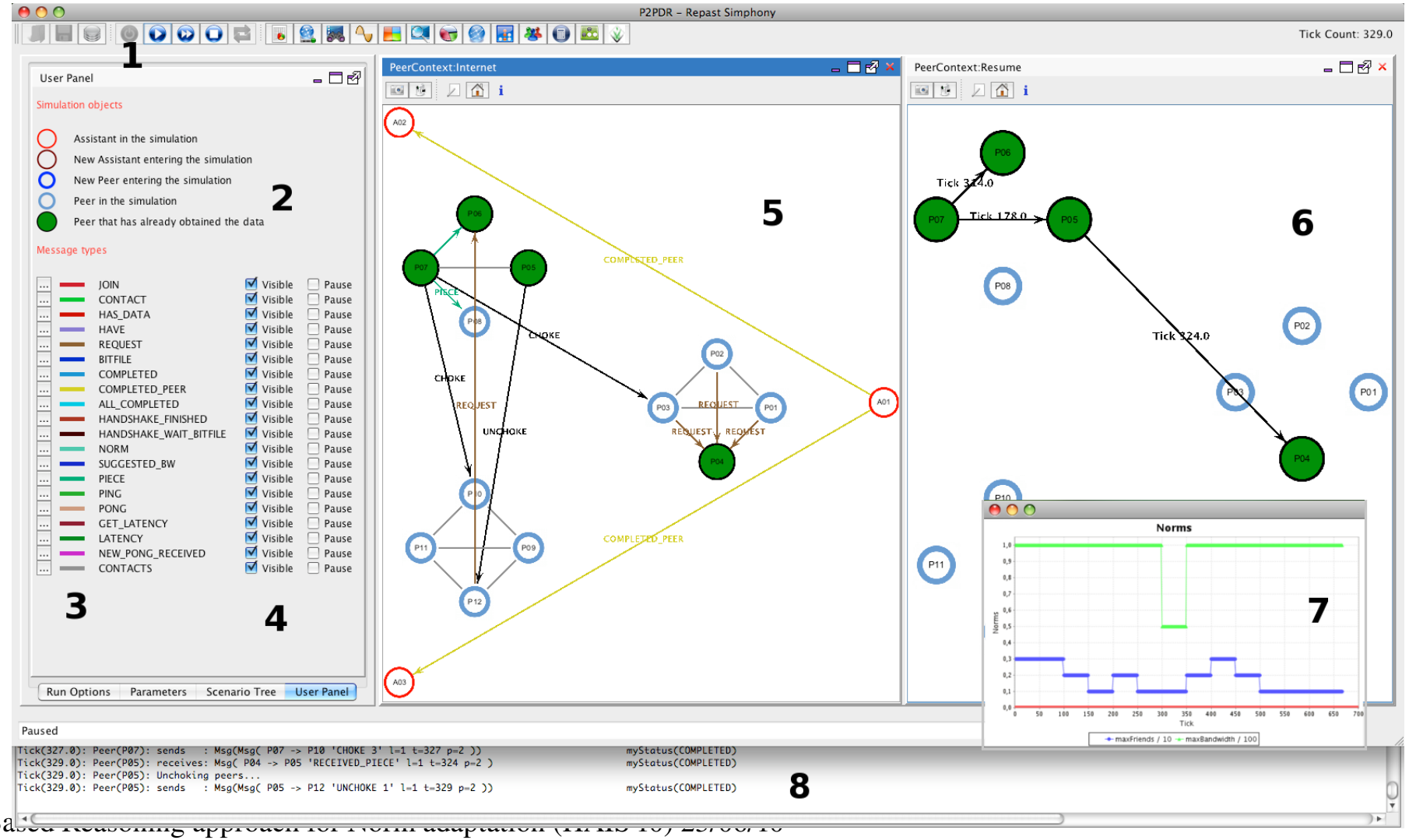
Current work:
Save also cases depending on evaluation
= Learning from own experience

in both cases:
CBR updates Case Base which may let it provide a different solution next time

Outline

- Motivation (OCMAS)
- Problem characterisation
- Case Study (P2P Sharing Network)
- Our approach (2-LAMA)
- Organisational Adaptation (norm adaptation)
- **Evaluation** (empirical, simulation)
- Conclusions and Future Work

- We have a **simulator** that let us **compare** different **implementations** in the P2P scenario:



Evaluation: results

$$\overline{\overline{BT}}$$

$$2L.a$$

$$2L.b$$

- **Sharing methods**:
  - **BT**: simplified standard Bittorrent protocol
    - 1 Tracker = agent's directory
    - all agents contact among them
    - at certain intervals, agents choose 3 previously interested agents to send data

$\overline{\overline{BT}}$

$2L.a$

$2L.b$

- **Sharing methods**:
  - **BT**: simplified standard Bittorrent protocol
  - **2L.a**: 2-LAMA without learning (only heuristic)
  - **2L.b**: 2-LAMA with CBR learning

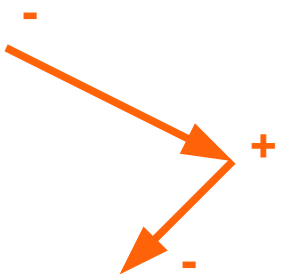
- Norms:
  - updated every 50 time units
  - Initial values: (equivalent to BT hardcoded restrictions)
    - maxBW=100%, maxFR=3

	$time$
BT	941.2
$2L.a$	834.9
$2L.b$	741.5

- **time** = time units to spread data among all agents
  - **2-LAMA** approaches (2L.a, 2L.b) improves time
    ↳ the time invested in communicating with ML is < benefits of having such an additional level.
  - **CBR learning** approach (2L.b) **improves** previous ones

Evaluation: results

	time	*cNet data*	*cML*	
BT	941.2	205344.1	11.0	-
2L.a	834.9	293526.7	35.9	5133.3
2L.b	741.5	292357.7	33.8	4694.1

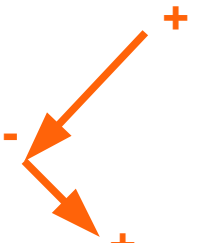


- **cNet** = the network cost consumed by all messages; $msg_{cost} = msg_{length} \times \#links_{traversed}$
  - 2-LAMA **requires more network**
    - more pieces of data messages sent (data)
    - DL2ML and ML2ML communications (cML)

  ↳ **but** it **avoids network saturation** and the corresponding delay → it presents shorter times.

Evaluation: results

	time	*cNet data*		*cML*	*h*
BT	941.2	205344.1	11.0	-	3.4
2L.a	834.9	293526.7	35.9	5133.3	2.9
2L.b	741.5	292357.7	33.8	4694.1	3.0



- **h** = the average number of links traversed by each message (hops)
  - **2-LAMA** has **more local communications** (i.e. intra-cluster) → convenient since local messages have lower latencies and costs
  - **CBR** learning (2L.b) reduces hops to increase locality but **not "too much"** as non-learning (2L.a) does.

Outline

- Motivation (OCMAS)
- Problem characterisation
- Case Study (P2P Sharing Network)
- Our approach (2-LAMA)
- Organisational Adaptation (norm adaptation)
- Evaluation (empirical, simulation)
- **Conclusions and Future Work**

- **2LAMA model** can deal with domains with the following feat.:
  - Non-task-decomposition,
    Non-fully observable,
    Non-deterministic,
    Dynamic, real-time,
    Run-time adaptation
- *in P2P scenario*:
  - 2LAMA improves BT performance
  - **CBR Learning improves** performance

Future Work:

- *Learning techniques*:

  - Reinforcement Learning

- *Open MAS issues*:

  - norm violations (related to self-interested competitiv. agents)

  - entering/leaving agents

Thanks for your attention

Questions**?**

Extra slides

Problem description

Problem characterisation

- **Environment**: [Russell&Norvig 95] [Wooldridge02]
  - non-task-decomposition oriented
    - **no** direct **mapping** between **goals** and **tasks**
    - → we are interested in using **norms to influence in agent behaviour** instead of assigning tasks

Problem characterisation

- **Environment**: [Russell&Norvig 95] [Wooldridge02]
  - non-task-decomposition oriented
  - non-fully observable
    - due to totally inaccessible information
    - due to privacy issues (locality)

Problem characterisation

- **Environment**: [Russell&Norvig 95] [Wooldridge02]
  - non-task-decomposition oriented
  - non-fully observable → non-deterministic

Problem characterisation

- **Environment**: [Russell&Norvig 95] [Wooldridge02]
  - non-task-decomposition oriented
  - non-fully observable → non-deterministic
  - dynamic
    - environment behaviour changes along time

Problem characterisation

- **Environment**: [Russell&Norvig 95] [Wooldridge02]
  - non-task-decomposition oriented
  - non-fully observable → non-deterministic
  - dynamic, real-time
    - an agent cannot deliberate for as long as desired to select its best course of action in a given scenario

Problem characterisation

- **Environment**: [Russell&Norvig 95] [Wooldridge02]
  - non-task-decomposition oriented
  - non-fully observable → non-deterministic
  - dynamic, real-time, run-time adaptation
    - there are structures that are adapted at the same time they are exploited

Problem characterisation

- Environment: [Russell&Norvig 95] [Wooldridge02]
  - non-task-decomposition oriented
  - non-fully observable → non-deterministic
  - dynamic, real-time, run-time adaptation
- **Agent population**:
  - self-interested, cooperative/competitive, developed by third-parties

Problem characterisation

- Environment: [Russell&Norvig 95] [Wooldridge02]
  - non-task-decomposition oriented
  - non-fully observable → non-deterministic
  - dynamic, real-time, run-time adaptation
- Agents population:
  - self-interested, cooperative/competitive, open

↳ there **exist** real **problems** with such **features**
e.g. a traffic scenario or a P2P sharing network

Results: network measures

	$time$	$cNet$
BT	941.2	205344.1
$2L.a$	834.9	293526.7
$2L.b$	741.5	292357.7

- **cNet** = the network cost consumed by all messages; $msg_{cost} = msg_{length} \times \#links_{traversed}$
  - 2-LAMA requires more network

Evaluation: results

	time	*cNet*	*data*
BT	941.2	205344.1	11.0
2L.a	834.9	293526.7	35.9
2L.b	741.5	292357.7	33.8

- **cNet** = the network cost consumed by all messages; $msg_{cost} = msg_{length} \times \#links_{traversed}$
  - 2-LAMA requires more network
    - more pieces of data messages sent (**data**)
      - CBR learning (2L.b) performs a better norm adaptation since it avoids some data cancels

Evaluation: results

	time	*cNet data*		*cML*
BT	941.2	205344.1	11.0	-
2L.a	834.9	293526.7	35.9	5133.3
2L.b	741.5	292357.7	33.8	4694.1

- **cNet** = the network cost consumed by all messages; $msg_{cost} = msg_{length} \times \#links_{traversed}$
  - 2-LAMA requires more network
    - more pieces of data messages sent (data)
    - DL2ML and ML2ML communications (**cML**)

CBR learning (2L.b) shorter sharing times save some adaptation cycles and their corresponding control messages. Also fewer cancels require less control messages.