Spatial Coordination of Pervasive Systems through Chemical-inspired Tuple Spaces

Mirko Viroli Matteo Casadei Sara Montagna Franco Zambonelli

Alma Mater Studiorum – Università di Bologna
{mirko.viroli,m.casadei,sara.montagna}@unibo.it

Università di Modena e Reggio Emilia
franco.zambonelli@unimore.it

Spatial Computing Workshop 2009

San Francisco
June 14, 2009

Summary

An emerging scenario

- “Eternally-adaptive” pervasive services [Villalba et al., 2008]
- Which infrastructure could support their coordination?
- This can be viewed as a spatial computing problem

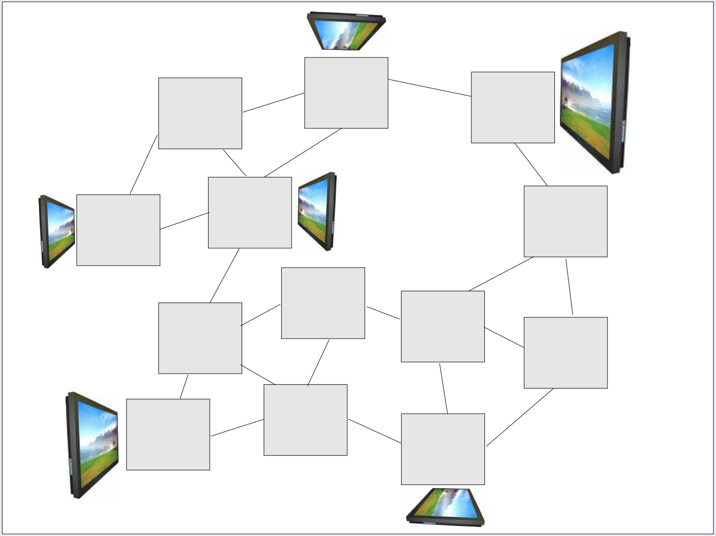
Proposing a model

- Chemical-inspired tuple space infrastructure
- Chemistry as inspiration for enacting self-* properties
- Spatial patterns naturally emerge

Outline

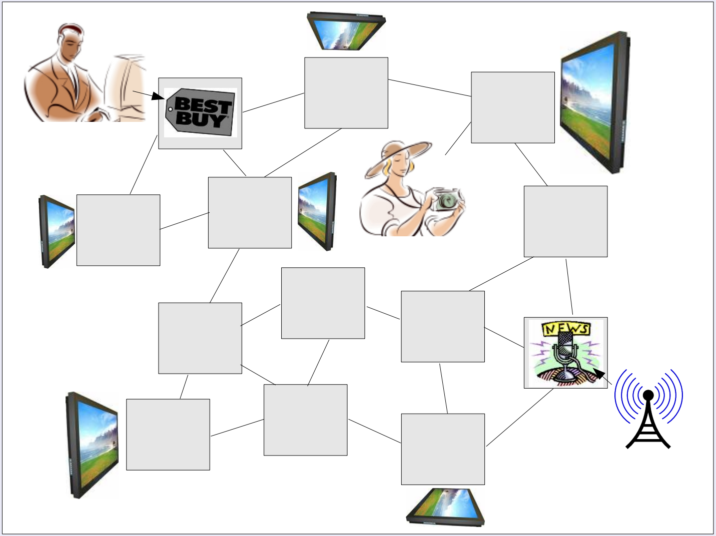
A scenario of pervasive displays

A pervasive network with displays spread around



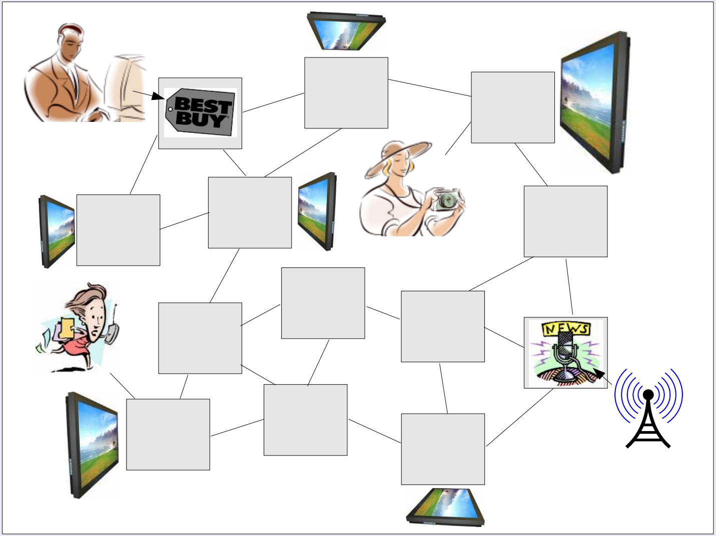
A scenario of pervasive displays

Visualisation services (news, ads, social data) get injected and diffuse



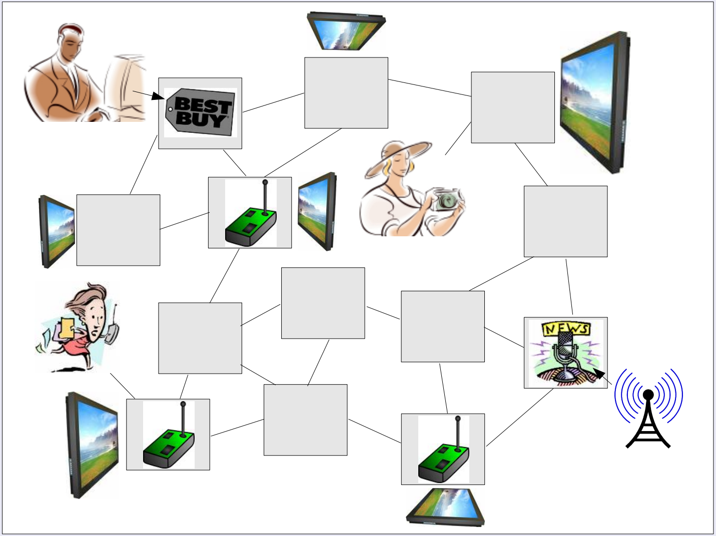
A scenario of pervasive displays

Displays should adapt visualisation to users nearby



A scenario of pervasive displays

Sensors provide contextual-information to improve adaptation



Framing the problem

Self-organising coordination of "Pervasive Services"

Software, data, devices, events are all to be seen as Pervasive Services, which:

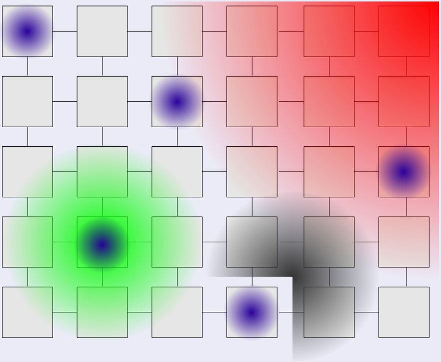
- are injected (or created) in some location, and may diffuse around
- compose, compete, interact with each other by matching
- are subject to context-dependency

Develop a middleware intrinsically surviving contingencies, such as:

- New visualisation services injected over time
- Displays and sensors are being continuosly mounted and unmounted
- Users enter the systems with unpredictable frequency and preferences

Pervasive services as a spatial computing problem

The middleware's job is to evolve the region of activity of services!



Local services

- User profiles
- Sensors' data
- Display devices

Diffused services

- Visualisation services
- "Signals" like computational fields

Research direction

Ingredients of the proposed framework

1. Distributed tuple-space infrastructure (likewise TOTA)
   - agents interact by inserting and retrieving tuples
2. Semantic tuples are used to represent the interface of pervasive services
   - sort of "live annotations" of pervasive services
   - what is important is their semantic content, not syntactic structure
3. Use a chemical metaphor for self-adaptive composition/evolution/diffusion of tuples (see [Viroli and Casadei, 2009])
   - very easy formal model (as Continuos Time Markov Chains)
   - can get inspiration from: natural biochemistry, artificial biochemistry, population dynamics

Outline

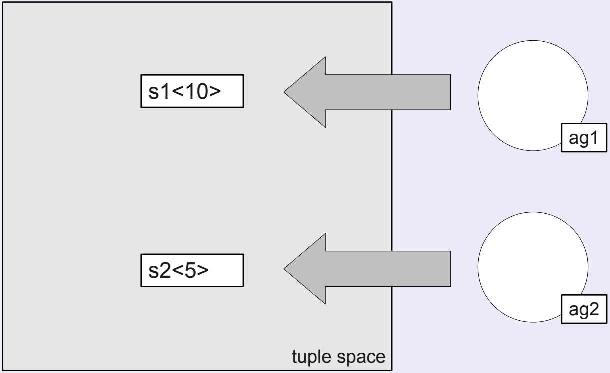
Biochemical Tuple Spaces

Main ideas w.r.t. a standard tuple space model

- Tuples have a concentration (a.k.a. weight, or activity value)
- Tuple spaces are equipped with chemical-resembling reactions
- Concentration is accordingly evolved "exactly" as in chemistry [Gillespie, 1977]
- Tuples are retrieved through semantic and fuzzy matching [Bobillo and Straccia, 2008]
- Some reactions can also fire a tuple from one space to another

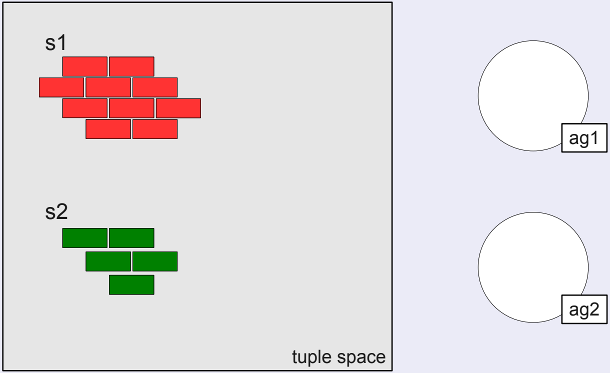
Tuple spaces

Inserting semantic tuples, with an initial concentration



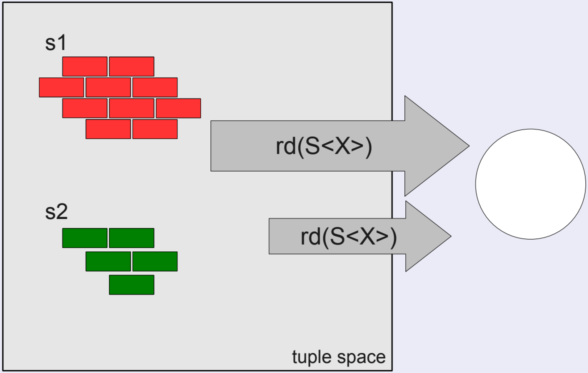
A pictorial representation

A tuple as substance of uniform molecules – but still a single tuple



Tuple retrieval

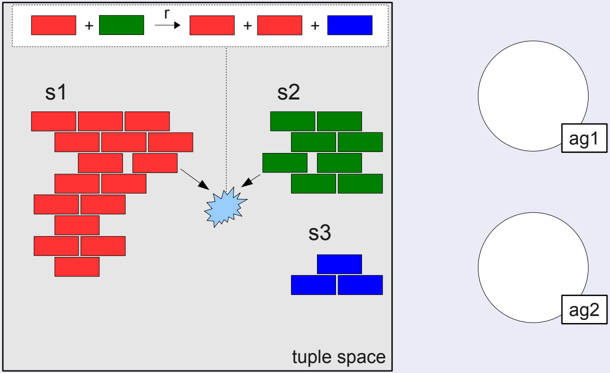
The higher the concentration, the easier is retrieval



Installing Chemical Reactions

A chemical reaction, with tuples in place of molecules

$$\mathtt{s1} + \mathtt{s2} \xrightarrow{r} \mathtt{s1} + \mathtt{s1} + \mathtt{s3}$$

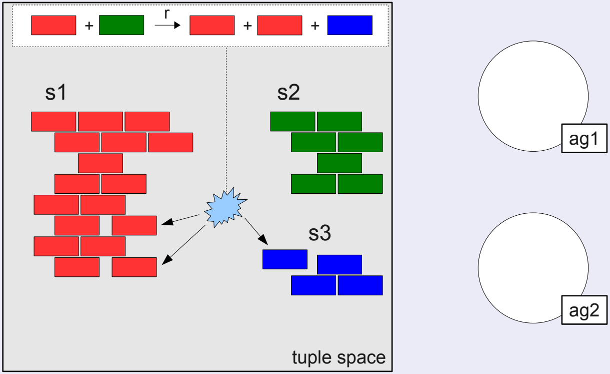


Firing Chemical Reactions

Reactions are executed over time according to [Gillespie, 1977]

$$\mathtt{s1} + \mathtt{s2} \xrightarrow{r} \mathtt{s1} + \mathtt{s1} + \mathtt{s3}$$

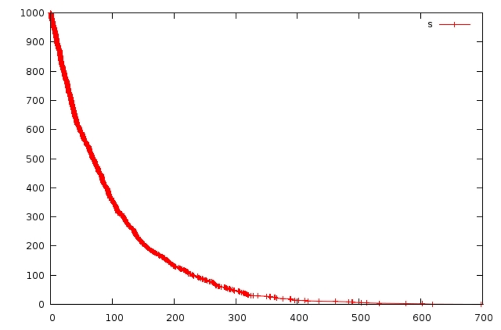
Transition (Markovian) rate: $r * \#\mathtt{s1} * \#\mathtt{s2}$



Decay example

After installing reaction $\mathtt{s} \xrightarrow{0.01} 0$

- We let tuple $\mathtt{s}$ decade (evaporate like pheromones)
- This is useful to enact time-pertinency
- It is perceived that the tuple is fading until disappearing
- E.g. $\mathtt{s}$ is a visualisation service with limited lifetime



On matching and rates

Our vision based on existing works in literature

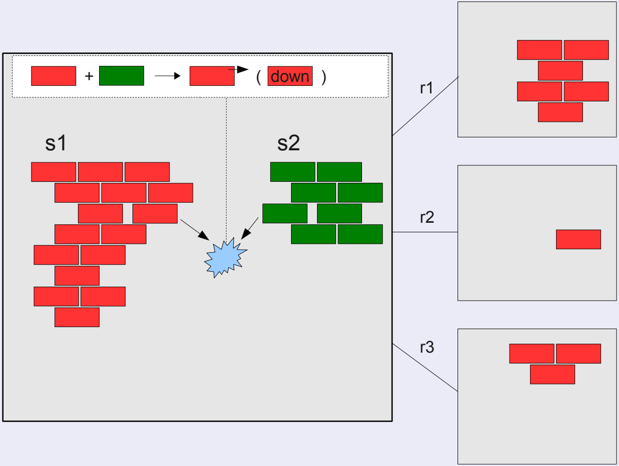
- Tuples represent individuals in Description Logic (DL)
- Tuple retrieval amounts to instance-test in fuzzy DL
- We use an application-dependent match function $\mu(t, t')$
  - yielding 0 is no match, 1 is perfect match, otherwise it is partial match
  - Chemical reactions are applied "modulo match ranking"
  - E.g. with $\mu = 0.5$, actual chemical rate is divided by 2
- A typical scenario of Web-based match-making (i.e. with preferences)

Example of general decay rule: $\mathtt{DECAY} \xrightarrow{r\_dec} 0$

- A specific tuple t decays with chemical rate $\mu(\mathtt{DECAY}, t) * r\_dec$
- It is μ that decides the actual decay speed (if any)

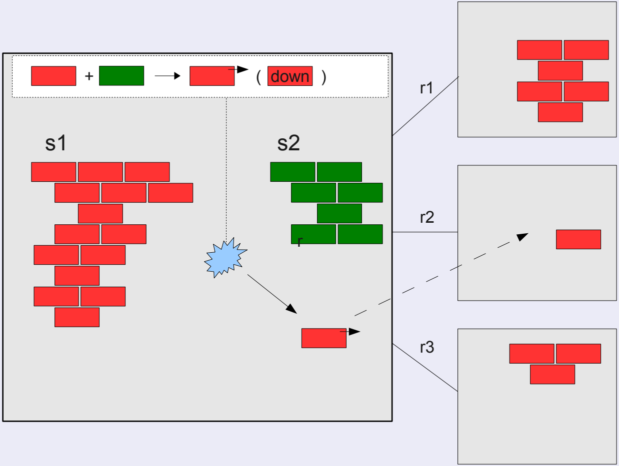
Tuple Transfer

Right-hand side of a reaction can have a "firing tuple"



From one node to a full network

Destination depends on link rates and gradient shape

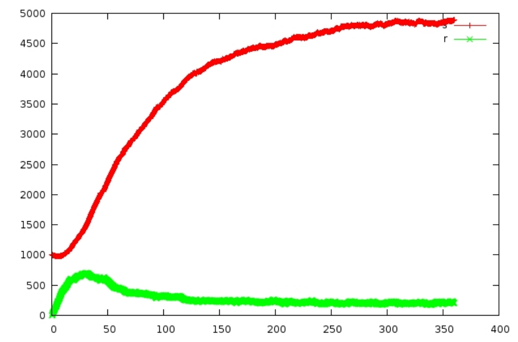


Feedback by using (a.k.a. prey-predator)

Idea: Matching Service-Request sustains the service

$$
\begin{array}{lrcl}
\texttt{(DECAY)} & \mathrm{SER} & \xmapsto{r\_dec} & 0 \\
\texttt{(USE)} & \mathrm{SER} + \mathrm{REQ} & \xmapsto{r\_use} & \mathrm{SER} + \mathrm{SER} + \mathtt{toserve}(\mathrm{SER}, \mathrm{REQ})
\end{array}
$$

Example simulation: $r\_dec = 0.01, r\_use = 0.00005, req\_rate = 50$



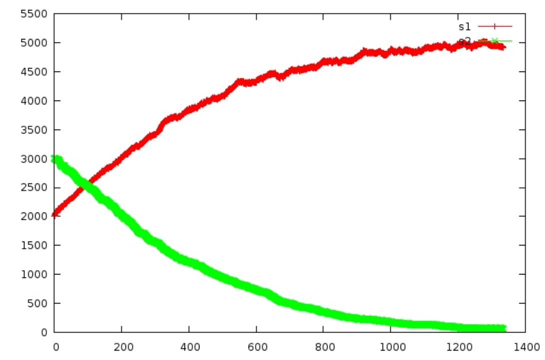
- Service level stabilises to $\frac{req\_rate}{r\_dec}$
- Use rate r_use influences reactiveness

Competition

What happens when more services can handle the same requests?

- higher concentration means higher match frequency
- some service may be a better match for requests

Example simulation: $r\_use_1 = 0.06$, $r\_use_2 = 0.04$



- The worst service loses competition and fades
- Supporting self-optimisation and self-adaptation..

Outline

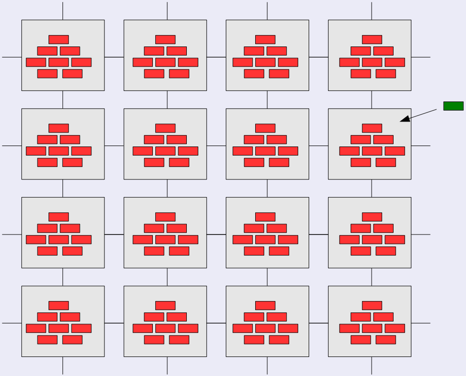
Spatial Diffusion and Competition

Initial scenario: a service monopolises a network and its requests

Services continuously diffuse around, by rule:

- Diffuse rule: $\mathtt{SER} \xrightarrow{r\_diff} \mathtt{SER}^{\rightsquigarrow}$

An unpredicted event: a better service is injected in a node

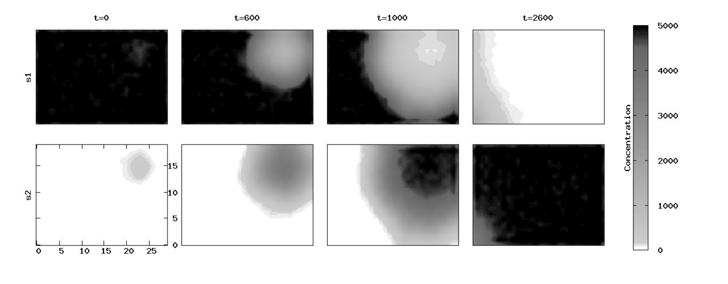


Resembling a biological tissue scenario

Simulation parameters

- 30x20 grid, requests arrive at rate 50
- old service use rate 0.05, new service use rate 0.1
- decay rate 0.01

Charts

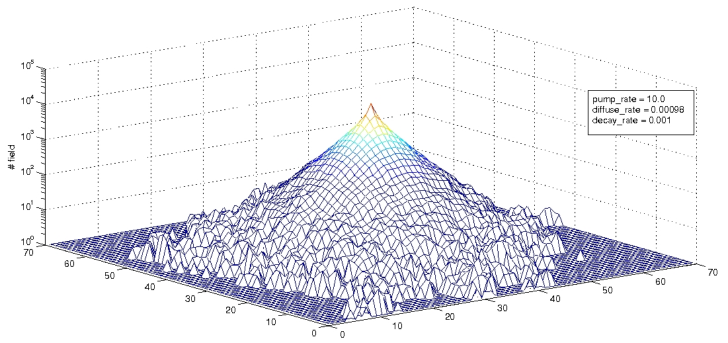


Computational field (a.k.a. gradient)

Chemical laws: a single PUMP token generates a FIELD

$$\begin{array}{lrcl}
\texttt{(PUMP)} & \texttt{PUMP} & \xmapsto{r_pump} & \texttt{PUMP} + \texttt{FIELD} \\
\texttt{(DECAY)} & \texttt{FIELD} & \xmapsto{r_decay} & 0 \\
\texttt{(DIFFUSE)} & \texttt{FIELD} & \xmapsto{r_diff} & \texttt{FIELD} + \texttt{FIELD}^{\rightsquigarrow}
\end{array}$$

Charts



Field ascent: a visualisation service searching a display

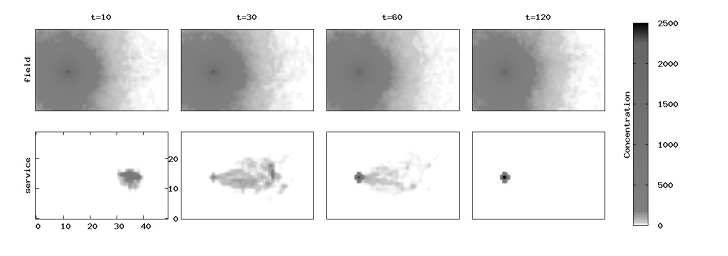
Ascent rule

$$(\mathtt{ASCENT}) \qquad \mathtt{REQ} \xmapsto{r\_asc} \mathtt{REQ}^{\leadsto}(\uparrow \mathtt{FIELD})$$

Simulation parameters

- Field: pump=10, decay=0.01, diffuse=0.0098
- Ascending service: initial conc=10′000, ascent rate=1.0

Charts



Path creation: a region for information transferring

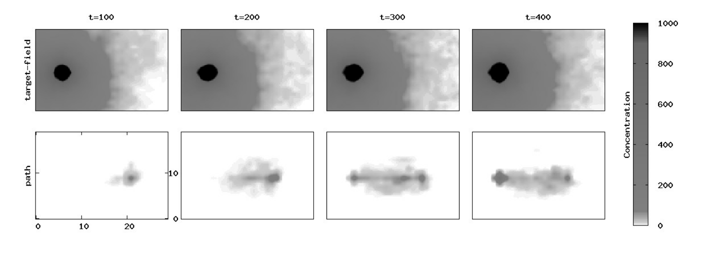
Path diffusion

$$\texttt{(DIFFUSE-P)} \qquad \texttt{PATH} \xmapsto{r\_diff} \texttt{PATH} + \texttt{PATH}^{\rightsquigarrow}(\uparrow \texttt{FIELD})$$

Simulation parameters

- Field: pump=10, decay=0.01, diffuse=0.0098
- Path: pump=10, decay=0.01, diffuse=0.0095

Charts

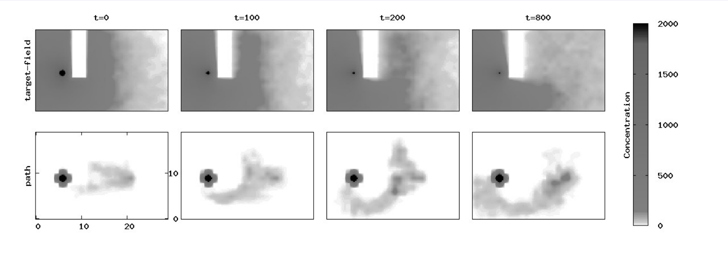


Self-adapting to topology and their changes

Dealing with obstacles

- A network failure
- The context suddenly inhibits fields

Charts

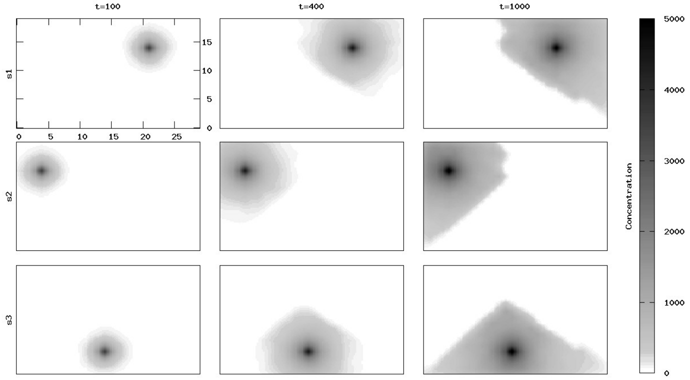


Segregation: splitting the network in contextual areas

Rule for segregated diffusion

$$\texttt{(SEGREGATE)} \quad \texttt{REGION} \xmapsto{r\_seg} \texttt{REGION} + \texttt{REGION}^{\rightsquigarrow}(\downarrow \texttt{OTHER\_REGIONS})$$

Charts



Conclusions: Ongoing Research

Implementation

- TuCSoN as underlying infrastructure
- Programmed to enact chemical behaviour [Viroli et al., 2009]
- Semantic tuple retrieval by Description Logics
- Studying performance issues

Applications

- Developing a protype for pervasive displays
- Evaluating other scenarios and other patterns

References I

Bobillo, F. and Straccia, U. (2008).
fuzzydl: An expressive fuzzy description logic reasoner.
In *2008 International Conference on Fuzzy Systems (FUZZ-08)*, pages 923–930. IEEE Computer Society.

Gillespie, D. T. (1977).
Exact stochastic simulation of coupled chemical reactions.
The Journal of Physical Chemistry, 81(25):2340–2361.

Villalba, C., Rosi, A., Viroli, M., and Zambonelli, F. (2008).
Nature-inspired spatial metaphors for pervasive service ecosystems.
In *Workshop on Spatial Computing*, Venice Italy.
Informal Proceedings.

References II

Viroli, M. and Casadei, M. (2009).

Biochemical tuple spaces for self-organising coordination.

In Field, J. and Vasconcelos, V. T., editors, *Coordination Languages and Models*, volume 5521 of *LNCS*, pages 143–162. Springer-Verlag.

11th International Conference (COORDINATION 2009), Lisbon, Portugal, June 2009. Proceedings.

Viroli, M., Casadei, M., and Omicini, A. (2009).

A framework for modelling and implementing self-organising coordination.

In Shin, S. Y., Ossowski, S., Menezes, R., and Viroli, M., editors, *24th Annual ACM Symposium on Applied Computing (SAC 2009)*, volume III, pages 1353–1360, Honolulu, Hawai'i, USA. ACM.

Spatial Coordination of Pervasive Systems through Chemical-inspired Tuple Spaces

Mirko Viroli Matteo Casadei Sara Montagna Franco Zambonelli

Alma Mater Studiorum – Università di Bologna
{mirko.viroli,m.casadei,sara.montagna}@unibo.it

Università di Modena e Reggio Emilia
franco.zambonelli@unimore.it

Spatial Computing Workshop 2009

San Francisco
June 14, 2009