Towards a Specification and Execution Environment for Simulations based on MMASS: Managing at-a-distance Interaction

Stefania Bandini, Sara Manzoni, Giuseppe Vizzari
DISCo - Università di Milano-Bicocca
{bandini,manzoni,viz}@disco.unimib.it

AT2AI – 2004 – April 16th 2004

Outline

- Context and motivations
- MMASS model – briefly
- General design and implementation issues/choices
- Focus on field diffusion
- Discussion
- Current/future works

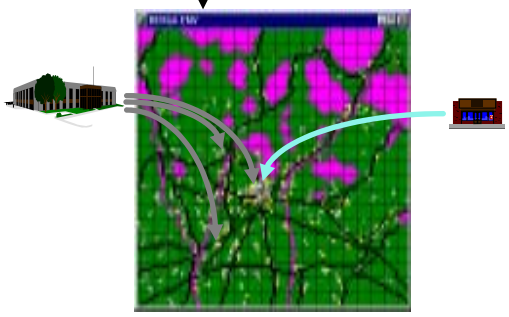
MAS applications: our experience (1)

Environmental modelling and planning

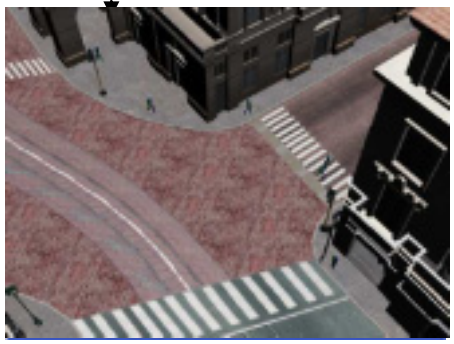
Crowd modelling and simulation

Biological systems

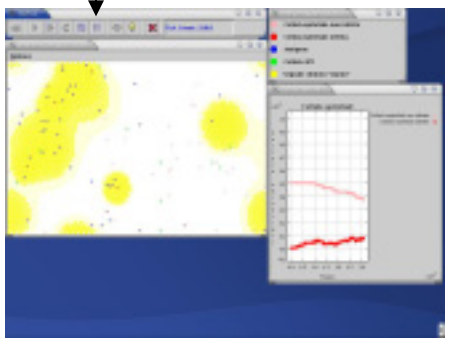
Context aware Systems (CSCW, other ongoing)



Cellular model



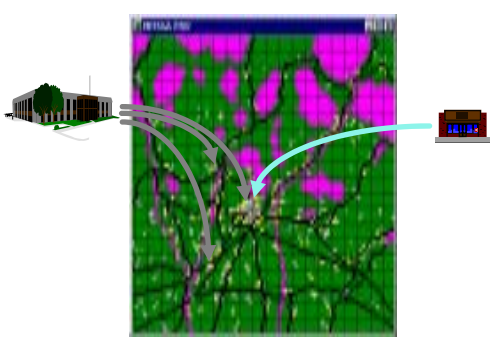
3D models and rendering tools

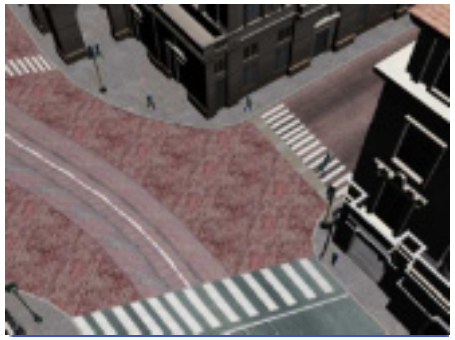


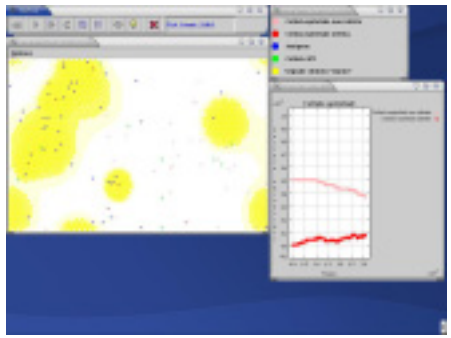
Repast MABS platform

- Models
- Platforms
- Middleware
- Methodo-logies
- Formalisms
- Languages
- …

MAS applications: our experience (2)







?

Common features

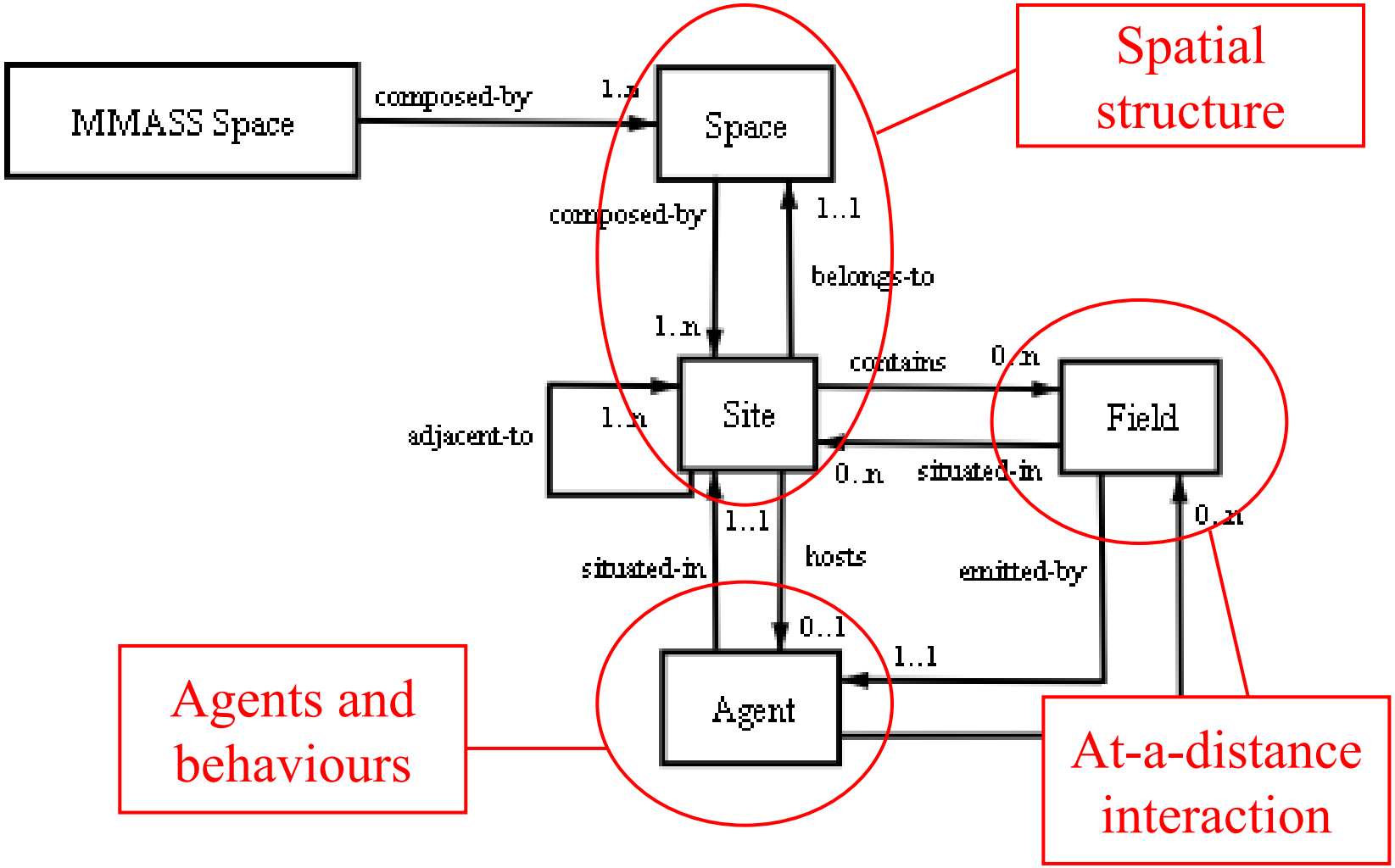
- Entities acting autonomously according to local point-of-view
- Heterogeneous and dynamic behavior
- Rich interaction model
- Spatial features of the environment
  - Key aspects of the applications
  - Influencing interaction

MMASS model

Supporting technologies

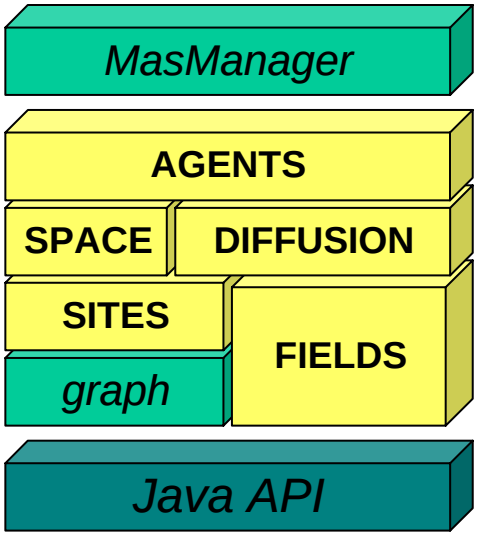
- Developed from scratch
- Adopted/adapted existing ones

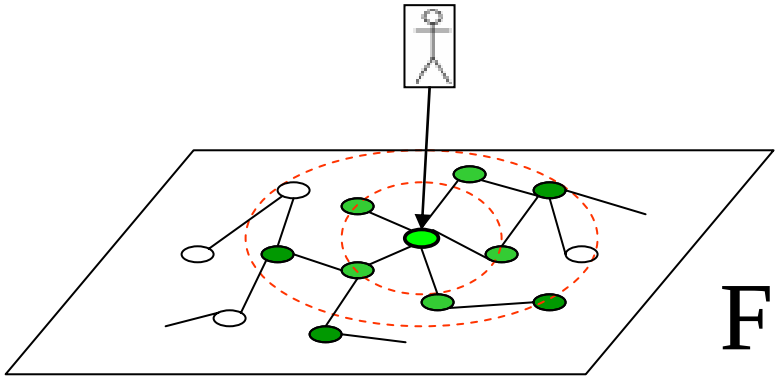
The MMASS Model in brief



Towards a platform for MMASS applications

- Language and related infrastructures to design, develop and run MMASS-based applications
- Set of components to manage
  - Space (layer definition, interaction …)
  - Agents (internal architectures, behavioral model …)
  - Interactions (between adjacent and at-a-distance entities)
- Tools to design, develop and deploy specific applications
- Internal development implies
  - Costs (time and resources to design, develop, test, and so on)
  - Insight and control (experience on various design/implem. issues, knowledge on adopted tools)





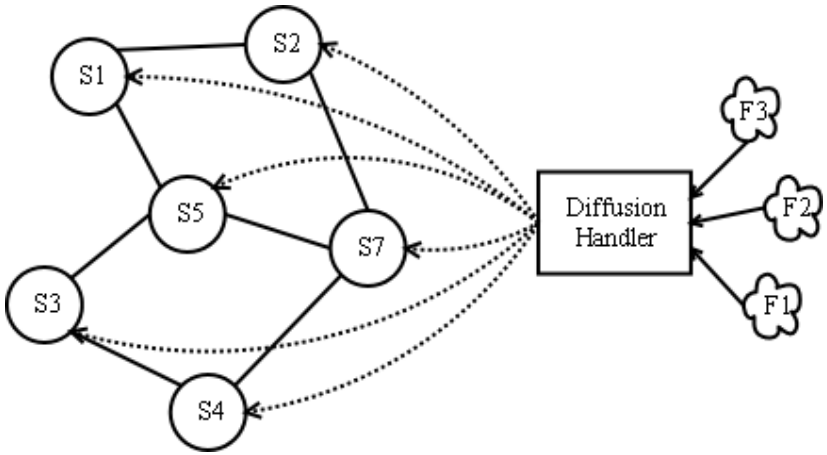
Focusing on Field-based Interaction

- Type of interaction
  - At-a-distance, space mediated, asynchronous interaction
  - Not based on an ACL, multicast, not message based
- How it occurs in MMASS?
  - ***Emission***: an agent emits a signal specifying its characteristic parameters (intensity, content, diffusion function)
  - ***Diffusion***: emitted signals spread throughout the spatial structure of the environment
  - ***Perception***: Agent state determines receptiveness and sensitivity
    - *Receptiveness* modulates field intensity (amplify or attenuate)
    - *Sensitivity* filters not perceivable signals (“low intensity”)

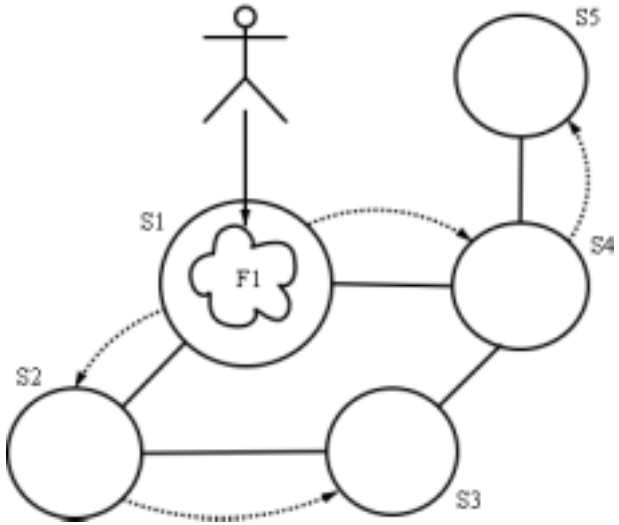
Issues in Field-based Interaction

- Issues in Field diffusion:
  - possibly irregular spatial structure
  - very general diffusion function
  - field composition (and possibly interference)
  - no assumption on system synchronicity
- “Non-functional” requirements
  - Time complexity
  - Memory occupation of the related structures

Centralized vs Decentralized approach

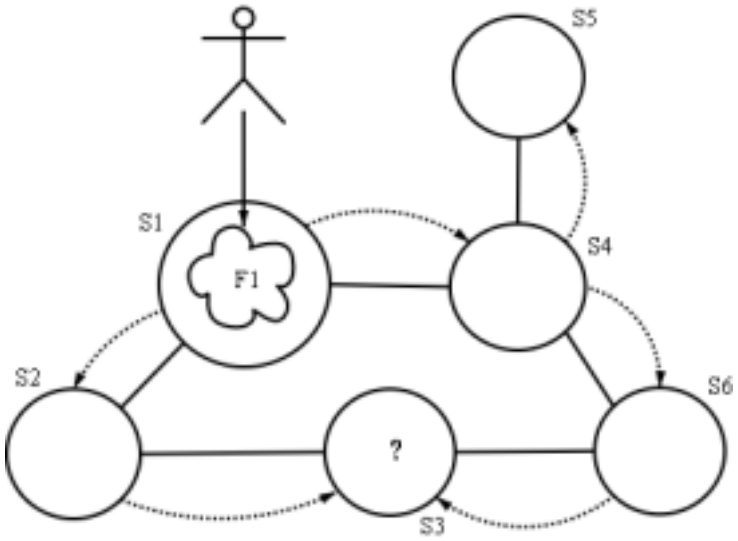


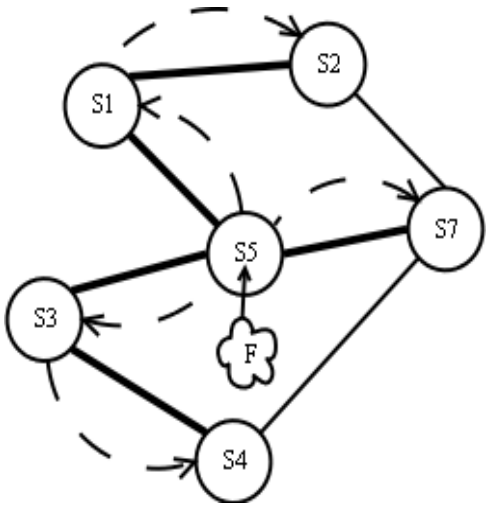
- Single diffusion handler
- Problems
  - Synchronization introduced (at handler level)
  - Possible bottleneck
- Possible solution for synchronous systems



- Sites or parts of the environment cooperate in order to perform diffusion
- Problems
  - Concurrent issues
  - Complex implementation
- The only solution for distributed implementation

Diffusion management





- Distributed algorithms (specifically, *routing*)
- Supporting structures
  - None (e.g. *flooding*)
  - Routing tables (e.g. *Link state*, *Distance vector*)

Field Replication and Diffusion paths

- According to application
  - Fields must reach destination sites once
  - Through the shortest path from their source
- This requires substantial modifications on flooding **or** different approaches providing some information related to the spatial structure for each site
- Every site needs to know
  - whether or not diffuse a field and through what edges
  - how its intensity should be modified (i.e. its distance from the emission site)

Structures supporting diffusion: MST

- Minimum Spanning Tree captures:
  - Minimal graph coverage (→ no fields replication)
  - Distance preservation from root (→ diffusion function can be easily computed)
- How many MSTs ?
  - One MST for every site (*minuses*: memory occupation, construction/upkeep costs; *plus*: in static-environment scenarios low diffusion cost)
  - Single MST and distance table (*minuses*: memory occupation is still $O(n^2)$; *pluses*: MST computed once for the whole space)
  - MSTs computed on-demand and stored in a limited size buffer (*minus*: diffusion operation can be more expensive; *plus*: memory occupation is linear)

Diffusion mechanisms summary

	Structures building complexity	Diffusion complexity	Memory occupation	Unique path for diffusion	Distance preservation	Comments
Flooding (corrected)	0	generally not known	0	no, correctives needed (field Id)	problematic (transient inconsistency)	termination is not obvious
MST for each site	$O(n + e)$	$O(\log_b n)$	$O(n^2)$	yes	yes (built via BSF)	
Single MST + distance table	$O(n + e)$	$O(n)$ (worst case)	$O(n^2)$	yes	yes	distance table could be a bottleneck in a distributed environment
MST cache	0	$O(\log_b n)$ $+ P(\overline{c}) \times O(n + e)$	$O(n)$	yes	yes	needs a cache management strategy

Discussion

- No generally optimal algorithm/strategy
- Application specific features might require/indicate specific solutions
  - In synchronicity assumption even centralized approaches might be feasible
  - Flooding might represent the best approach if field replication is required (or at least, not a serious issue)
- Application could require a richer spatial model
  - How to handle weighted edges and labeled sites?

Current/Future works (1)

- Application of present approaches and implementation to various domains
  - Crowd simulation in 3D environments, with industrial partners
  - Urban planning within the Exystence network
- Adoption of the interaction model for the development of context-aware applications
  - Context dependant service provisoning (e.g. touristic and automotive domain)
  - Promoting awareness in collaborative environments – CSCW domain

Current/Future works (2)

- What about the platform?
  - Some elements were developed and exploited (field diffusion)
  - Other ones are still in early stages of development or just basic strategies were developed (e.g. concurrecy issues)
  - Analysis, design and evaluation of further platform elements and improvements will proceed in parallel with current/future applications and projects
  - A long term goal of the platform is to allow non-programmers to use the platform, for instance to design and execute simulations in specific application domains

Giuseppe Vizzari
Dipartimento di Informatica, Sistemistica e Comunicazione
Università degli Studi di Milano-Bicocca
viz@disco.unimib.it

Images of Scala Square appear courtesy of GeoSim Systems

Thank you!