INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Agent-based Modeling and Simulation

Michal Jakob
Agent Technology Center, Department of Cybernetics
Czech Technical University in Prague

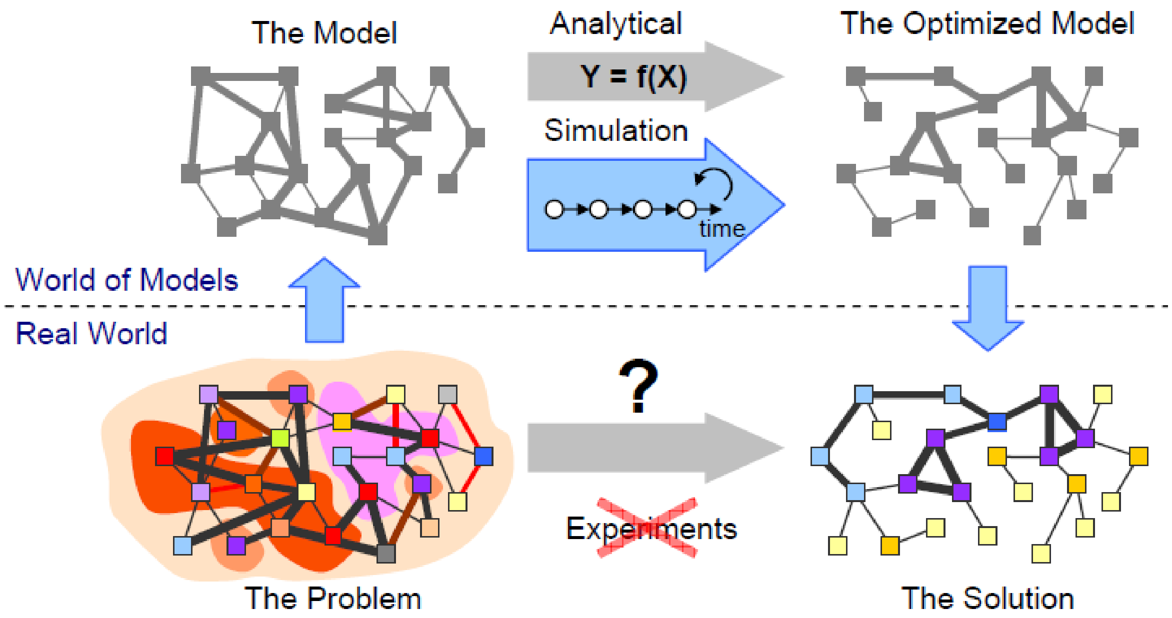
22 March 2010

Tato prezentace je spolufinancována Evropským sociálním fondem a státním rozpočtem České republiky.

Introduction

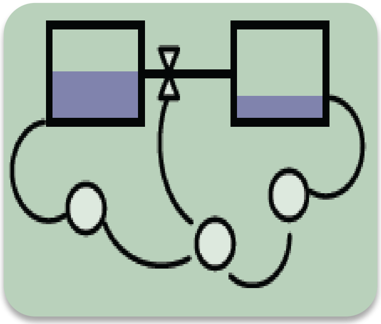
Introduction

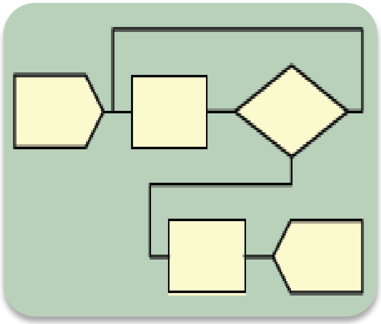
Simulation is a key tool for obtaining **insight** and **foresight** regarding the operation of complicated systems

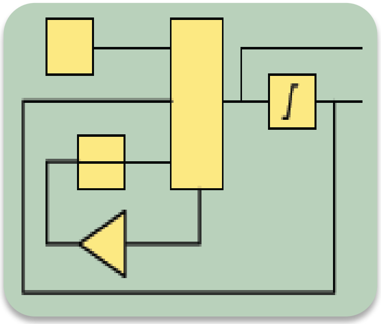


From: Borshchev, A. et al (2004): From system dynamics and discrete event to practical agent based modeling: Reasons, techniques, tools

S&M Approaches







System dynamic

- states, feedbacks and delay structures
- continuous
- global, aggregate view

Discrete Event

- entities and resources
- discrete, event-based
- global entity processing algorithm

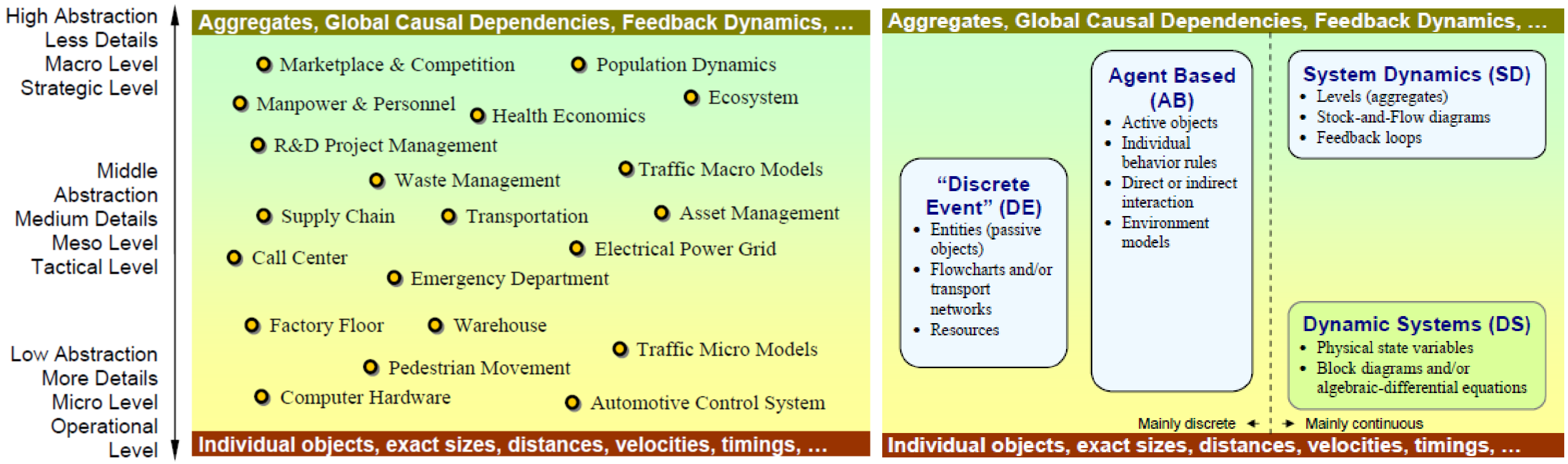
Agent Based

- active entities and the environment
- decentralized, individual perspective
- **global behavior emerges**

Dynamic Systems

- state variables and differential equations
- direct physical meaning, no aggregation

Levels of Abstraction



From: Borshchev, A. et al (2004): From system dynamics and discrete event to practical agent based modeling: Reasons, techniques, tools

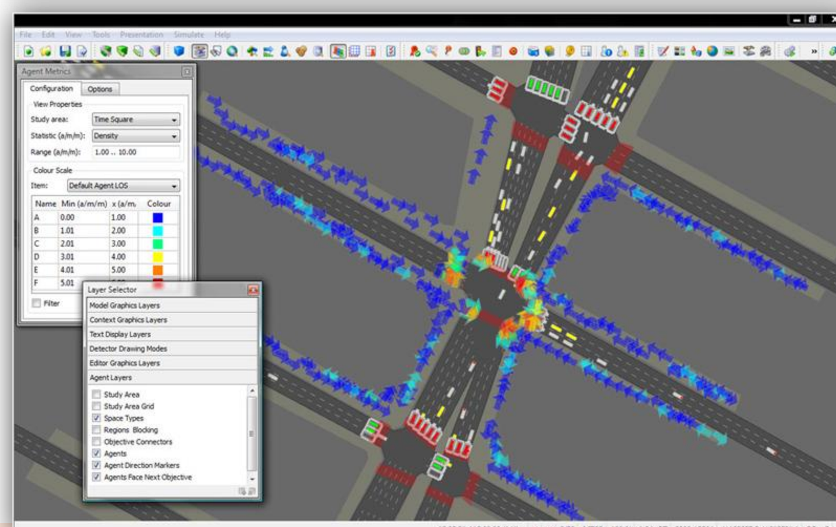
Illustrative Example

Pedestrian simulation

- Each pedestrian modeled as an agent sensing the environment and interacting with other pedestrian agents

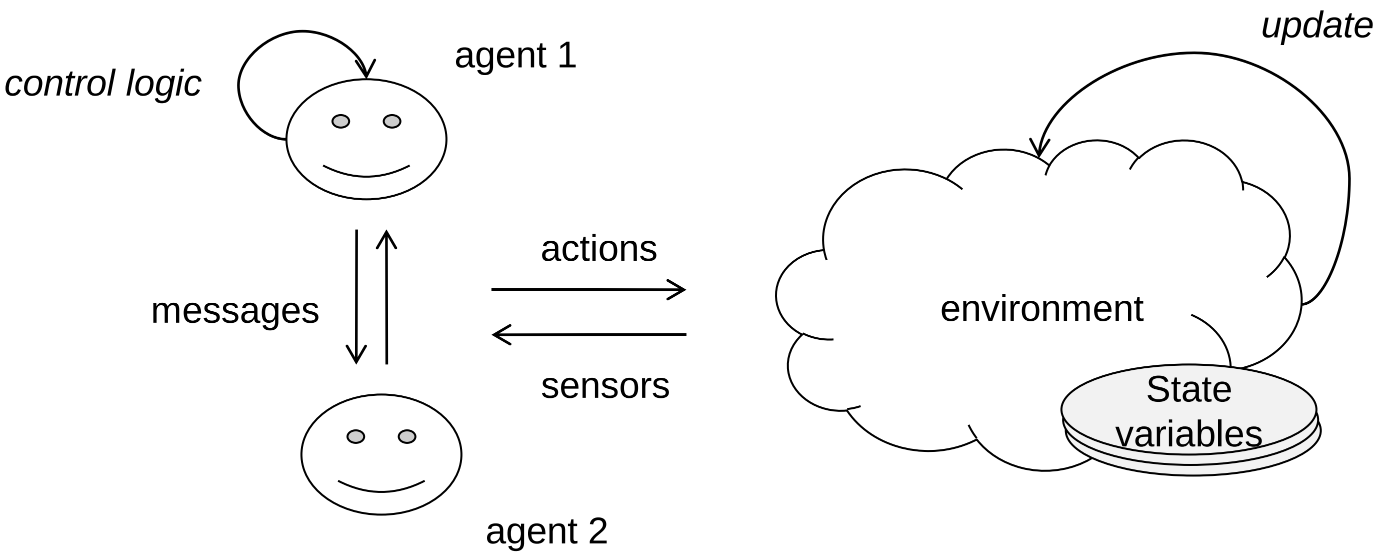
The model allows to

- determine crowd flows and densities under various scenarios
- optimize crowded public spaces for capacity, comfort and safety



Architecture of the Agent-based Simulation

Structure of the Agent-based Simulation



Agents drive the model through local behaviors and direct and indirect interaction with each other and with the environment

Environment state is modified by agent actions and/or agent-independent processes (e.g. weather)

Agent

Key component of an-agent based model

Proactive entity acting towards its objective

Key operations

- processing sensory inputs
- selecting actions to perform
- managing communication

Additional operations

- adaptation
- deliberation

Agent's Control Logic

BDI (belief-desire-intention) architecture as a foundation

Simple control logic implementations used in practice

- (hierarchical) finite state machines
- rule engines
- scripts

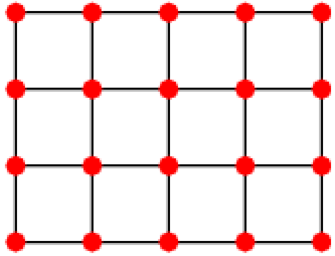
Calls to special-purpose subsystems

- navigation
- environment perception

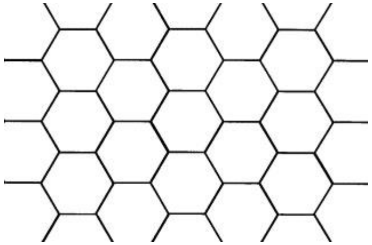
Environment

Shapes agent's operation, mediates inter-agent interaction, can impact inter-agent communication

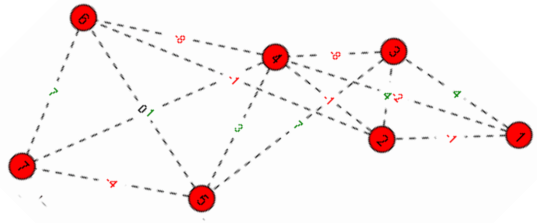
Abstract environments



rectangular grid



hexagonal grid



general graph

Realistic/VR: open landscape, urban, indoor

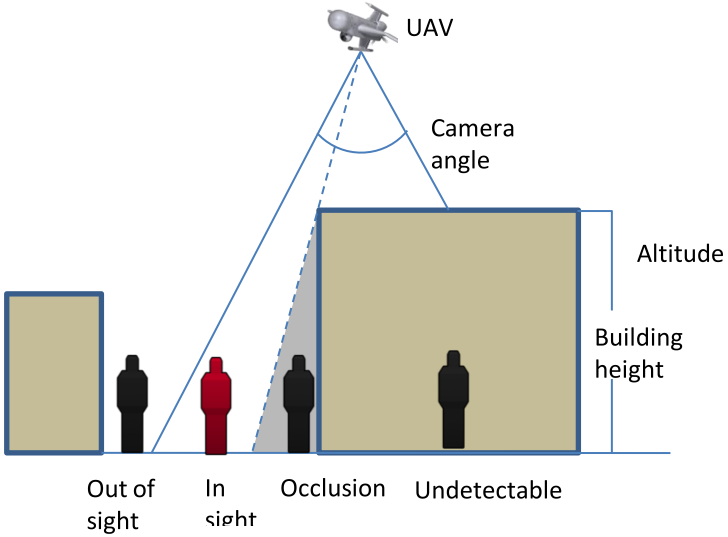
Sensors

Enables the agent to access the environmental state

- low-level – direct perception (e.g. image from a camera)
- high-level – interpreted scene (e.g. walls, people)

Efficient implementation crucial in more complex environments

- partitioning
- caching



example sensor model

Developing agent-based simulations

Development Cycle

Design

- identify agents and environmental features
- choose the right modeling abstraction critical

Implementation

- agent's control logic
- environmental features and dynamics

Calibration and Validation

- micro-level and macro-level
- real-world data-based (if possible)

Calibration and Validation

Micro-level

- parameter correspondence (e.g. size, speed, visual range, …)
- behavior correspondence (reaction under different circumstances)

Macro-level

- compare outcome distributions
- comparison with real system traces (start from the same initial conditions)

Generally very difficult due to inherent complexity of the system

Platforms and Tools

General platforms still only in an early stage

- academic: RePast, NetLogo
- commercial: AnyLogic
- A-Globe Simulation

Special-purpose platforms more mature

- traffic modeling: AIMSUN
- pedestrian modeling: LEGION

GIS tools and data sources

- Google Earth, NASA WorldWind
- http://www.openstreetmap.org/

Advantages

Captures **emergent phenomena**

- the whole is not a sum of parts – system cannot be abstracted
- non-linearity, discontinuity, phase-transitions

Provides a natural description of the system

- agents easy to identify in the real-world system
- agents’ activities easier to describe than global processes

Flexible

- agents’ behaviors can be easily extended/specialized
- size of the model can be scaled

When to Use

Complex interaction between entities

- interaction topology not fixed
- feedback relationships, network effects

Heterogeneity of entities

- each agent has a different behavior

Behavior of individuals is complex

- learning, adaptation

Unknown global process structure and/or parameters

Environment is crucial and agents are mobile

Application Areas

traffic and transport

- optimization of traffic networks, understanding and eliminating congestion

pedestrians and crowds

- capacity optimization, evacuation procedures

organizations

- organization design optimization, operation risk estimation

markets and economies

computer networks

- bandwidth usage estimation, worm infection modeling

security

- crime modeling, vulnerability estimation

Conclusions

Conclusions

Most recent addition to modeling and simulation toolbox

Bottom-up approach

Most suitable for complex systems composed of autonomous, interacting entities

Allows high-fidelity models at the expense of high-computational cost

Mature tools exist for specific domains (e.g. transport, crowds)

General purpose platforms and tools still under development