D. Moser, A. Riener, K. Zia, A. Ferscha
*Department for Pervasive Computing, JKU Linz/Austria*

# “Parallel Simulation of Social Agents using Cilk and OpenCL”

**DS-RT 2011**
15th International Symposium on Distributed Simulation and Real Time Applications
September 4,-7, 2011, Salford/Manchester, UK

SOCIONICAL
This work is supported under the FP7 ICT FET program of the European Commission under grant agreement No 231288 (SOCIONICAL)

**Dr. Andreas Riener**
JKU Linz, Department for Pervasive Computing
Altenberger Straße 69, A-4040 Linz
www.pervasive.jku.at/about_us/staff/riener

# Socio-Technical Systems: **Motivation for Modeling/Simulation**

"Development of complexity science based modeling, prediction and simulation methods for large scale socio-technical systems in an AmI based smart environment"

- **Experiments**: standard way of collecting evidence in such (dynamic) systems
  - allow to analyze situations, person behavior and to interview test participants

- **Experimentation** is, however, **not possible** in large social systems
  - undesirable/unaccepted
  - repeatability not given
  - dangerous (for involved persons, infrastructure)
  - impossible (in terms of scale or behavior)
    (i) evacuation of a large megacity with million of peoples is not possible
    (ii) different behavior of entities/persons on artificial/simulated hazards compared to a real incident; generating of a "real nuclear incident" is not feasible

- **Solution**: Simulated interaction of agents based on realistic behavioral rules
  - "agent" = entity with realistic behavior and interaction capabilities
  - AmI technology to "enhance" agents (FOV, knowledge, etc.)
  - real underlying space model

## Socio-Technical Systems: **Motivation for Modeling/Simulation (2)**

Agent-based modeling (ABM)

- A **widely used** analytical method capable to represent individual entities and their interactions [Gilbert2008]
- **Resource intensive** – using a single machine, simulation of only small models/local behavior possible
- Only suited for **small- to medium-sized problems** (single workstation) *[Zia2010]*

Discrepancy “resources” ↔ “large scale”?

- Due to advancements in processing power (GPGPU) and/or cluster technology (PDS) no longer a problem…

Close-to-reality results

- Developments in cognitive social modeling allows for the first time for close-to-reality simulation of social or collective phenomena (e.g., group formation)
- Further model up-scales allows ABM to explain the emergence of higher order patterns (movement dynamics in traffic jams, behavioral patterns in global social networks, social segregation across populations)

# Socio-Technical/Social Systems: **Agent-Based Modeling**

## Designing agents in software
## (*individual representation of an agent*)

- **Perception**: Agents can perceive their neighborhood, i.e. they can determine what agents (including space agents) are in their vicinity
- **Performance**: How do they perform their activity which may include motion, communication (interaction) and action (changing states of itself or other agents)
- **Memory**: They have a memory where they can record their action and states which may include the history
- **Policy**: they have a set of rules, heuristics, or strategies that determines, given their present situation and their history, what behaviors they would now carry out

```
for each agent a in simulation
stats = synchronize(a);

if AmI-assisted agent update-intentions
else update-proximity-parameters

for each exit e
dist = decision-param (e); distance/belief etc.
hope_a(e)= … ;based on group emotions, individualism
fear_a(e)= … ;based on group emotions, individualism
attract_a(e)= …;based on group emotions, individual…

curr-exit = choose exit with max attract
curr-dir = get-direction_curr-exit ;floor field
heading = curr-dir ; setting heading of the agent
MOVE

update-proximity-parameters
For each agent n in neighborhood of a
for each exit e
update group fear, hope and attract

update-intentions
update-proximity-parameters
For each agent n in the neighborhood of a
belief_n(curr-exit)= … ;based on belief/trust of group
trust_n(a)= … ;based on belief of a on curr-exit
```

# Socio-Technical/Social Systems: **Agent-Based Modeling**

## Designing agents in software

- **Perception**: Agents can perceive their neighborhood, i.e. they can determine what agents (including space agents) are in their vicinity
- **Performance**: How do they perform their activity which may include motion, communication (interaction) and action (changing states of itself or other agents)
- **Memory**: They have a memory where they can record their action and states which may include the history
- **Policy**: they have a set of rules, heuristics, or strategies that determines, given their present situation and their history, what behaviors they would now carry out

```
for each agent a in simulation
    stats = synchronize(a);

if AmI-assisted agent update-intentions
    else update-proximity-parameters

        for each exit e
    dist = decision-param (e); distance/belief etc.
    hope_a(e)= … ;based on group emotions, individualism
    fear_a(e)= … ;based on group emotions, individualism
    attract_a(e)= …;based on group emotions, individual…

curr-exit = choose exit with max attract
curr-dir = get-direction_curr-exit ;floor field
heading = curr-dir ; setting heading of the agent
MOVE

update-proximity-parameters
For each agent n in neighborhood of a
    for each exit e
    update group fear, hope and attract

update-intentions
update-proximity-parameters
For each agent n in the neighborhood of a
  belief_n(curr-exit)= … ;based on belief/trust of group
  trust_n(a)= … ;based on belief of a on curr-exit
```

# Socio-Technical/Social Systems: **Agent-Based Modeling**

## Designing agents in software

- **Perception**: Agents can perceive their neighborhood, i.e. they can determine what agents (including space agents) are in their vicinity
- **Performance**: How do they perform their activity which may include motion, communication (interaction) and action (changing states of itself or other agents)
- **Memory**: They have a memory where they can record their action and states which may include the history
- **Policy**: they have a set of rules, heuristics, or strategies that determines, given their present situation and their history, what behaviors they would now carry out

```
for each agent a in simulation
  stats = synchronize(a);

if AmI-assisted agent update-intentions
else update-proximity-parameters

for each exit e
  dist = decision-param (e); distance/belief etc.
  hope_a(e)= … ;based on group emotions, individualism
  fear_a(e)= … ;based on group emotions, individualism
  attract_a(e)= …;based on group emotions, individual…

curr-exit = choose exit with max attract
curr-dir = get-direction_curr-exit ;floor field
heading = curr-dir ; setting heading of the agent
MOVE

update-proximity-parameters
For each agent n in neighborhood of a
  for each exit e
    update group fear, hope and attract

update-intentions
update-proximity-parameters
For each agent n in the neighborhood of a
  belief_n(curr-exit)= … ;based on belief/trust of group
  trust_n(a)= … ;based on belief of a on curr-exit
```

# Socio-Technical/Social Systems: **Agent-Based Modeling**

## Designing agents in software

- **Perception**: Agents can perceive their neighborhood, i.e. they can determine what agents (including space agents) are in their vicinity
- **Performance**: How do they perform their activity which may include motion, communication (interaction) and action (changing states of itself or other agents)
- **Memory**: They have a memory where they can record their action and states which may include the history
- **Policy**: they have a set of rules, heuristics, or strategies that determines, given their present situation and their history, what behaviors they would now carry out

```
for each agent a in simulation
  stats = synchronize(a);

if AmI-assisted agent update-intentions
else update-proximity-parameters

  for each exit e
  dist = decision-param (e); distance/belief etc.
  hope_a(e)= … ;based on group emotions, individualism
  fear_a(e)= … ;based on group emotions, individualism
  attract_a(e)= …;based on group emotions, individual…

curr-exit = choose exit with max attract
curr-dir = get-direction_curr-exit ;floor field
heading = curr-dir ; setting heading of the agent
MOVE

update-proximity-parameters
For each agent n in neighborhood of a
  for each exit e
  update group fear, hope and attract

update-intentions
update-proximity-parameters
For each agent n in the neighborhood of a
  belief_n(curr-exit)= … ;based on belief/trust of group
  trust_n(a)= … ;based on belief of a on curr-exit
```

# Socio-Technical/Social Systems: **Agent-Based Modeling**

## Designing agents in software

- **Perception**: Agents can perceive their neighborhood, i.e. they can determine what agents (including space agents) are in their vicinity
- **Performance**: How do they perform their activity which may include motion, communication (interaction) and action (changing states of itself or other agents)
- **Memory**: They have a memory where they can record their action and states which may include the history
- **Policy**: they have a set of rules, heuristics, or strategies that determines, given their present situation and their history, what behaviors they would now carry out

```
for each agent a in simulation
  stats = synchronize(a);

if AmI-assisted agent update-intentions
  else update-proximity-parameters

    for each exit e
  dist = decision-param (e); distance/belief etc.
  hope_a(e)= … ;based on group emotions, individualism
  fear_a(e)= … ;based on group emotions, individualism
  attract_a(e)= …;based on group emotions, individual…

curr-exit = choose exit with max attract
curr-dir = get-direction_curr-exit ;floor field
heading = curr-dir ; setting heading of the agent
  MOVE

update-proximity-parameters
For each agent n in neighborhood of a
  for each exit e
  update group fear, hope and attract

update-intentions
update-proximity-parameters
For each agent n in the neighborhood of a
  belief_n(curr-exit)= … ;based on belief/trust of group
  trust_n(a)= … ;based on belief of a on curr-exit
```

# Socio-Technical/Social Systems: **Agent-Based Modeling**

## Behavioral/social features of agents

- **Autonomy:** Ability to make its own decisions without a central controller
- **Social ability:** Ability to interact with other agents
- **Reactivity:** Ability to react to a stimulus
- **Proactivity:** Ability to pursue its goal on its own initiative
- **…and more capabilities/ requirements for social agents with individual behavior**
  - heterogeneity
  - space mapping
  - behavioral adaptation/learning
  - etc.

## Approaches for Distributed/Parallel Computing

### Hardware

*Multithreaded parallel computing (similar CPU cores)*

*Heterogeneous platforms of CPUs, GPUs, and other processors*

### Software tools/frameworks

*Cilk*

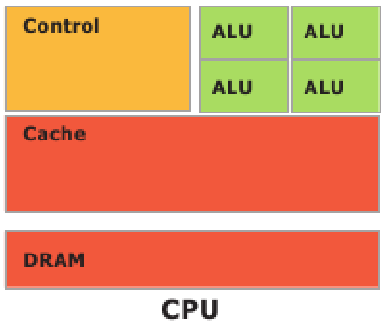
*CUDA/OpenCL*

# High Performance Simulation on Large Scale: **HW/SW Setting**

## A) Hardware

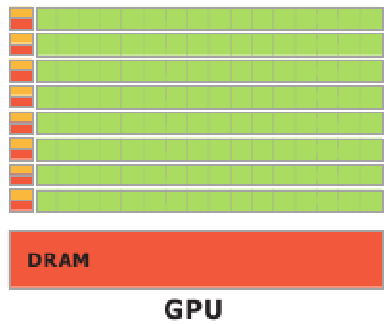
### Multi-CPU System (*shared memory, SMA*)

- Altix 4700 (SGI) - 64 Blades
- 128 Intel Itanium2 Montecito CPU's (1.6GHz, 18MB L3, Dual Core)
- 1 TB RAM (16GB per Blade)
- 24 x 300GB SAS HDD
- Network (2 x 1GigaBit,10GigaBit)
- Architecture:




### Parallel GPU System

- NVidia GeForce 9700M GT
- G96 PU (625MHz)
- 32 stream processors
- 512MB GDDR3 (800MHz, 256bit)
- DirectX 10, Shader 4.0
- 504 Mio. Transistors, 65 nm
- Architecture:




## B) Software

### Cilk

- Framework to run multi-threaded programs on shared-memory machines
- C/C++ extension
- Only two keywords "*spawn*", "*sync*"
- Integrated scheduler capable to distribute the workload
- Designed by MIT (from 1994), now maintained by INTEL

### OpenCL

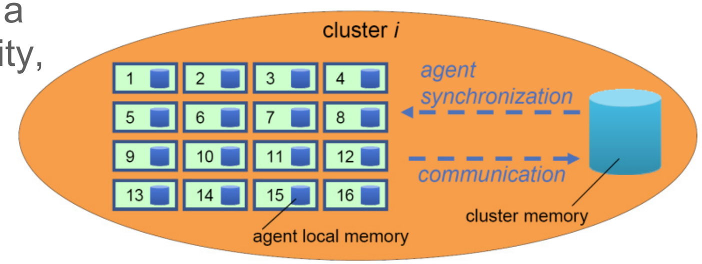
- Framework to develop generalized parallel executable programs
- Standardized language extension for C/C++ used with CPU's and GPU's
- Designed by Apple, now maintained by the Khronos group (Apple, Intel, AMD, etc.)

# High Performance Simulation on Large Scale: **Simulation Conditions**

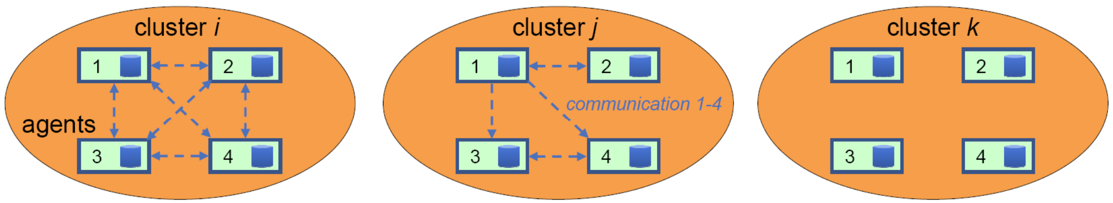
## Model characteristics, execution parameters

- **Models:** (i) abstract cluster behavior, (ii) realistic individual behavior model
- **Simulation scale:** $10^6$, $10^7$ agents
- **Execution:** shared memory cluster (*Cilk*), GPU (*OpenCL*); 5 repetitions each, run time per cycle up to *136min.* (single core), *3.5min.* (100 cores), *53sec.* (GPU)
- **Model behavior variation:**

(i) cluster size: number of agents abstracted into a single cluster (group agents with same functionality, e.g. cognitive behavior, communication abilities, etc.); similar for all clusters; range 2-32 ($2^1$ to $2^5$); example cluster size 16: $10^6/16$=62,500 clusters




(ii) connectivity: communication rate of agents in a cluster (FOV, Wifi range, etc.); similar for all clusters; range 0-1 (steps of 0.2); e.g. connectivity 1.0, 0.5 and 0 in a cluster of 4 agents



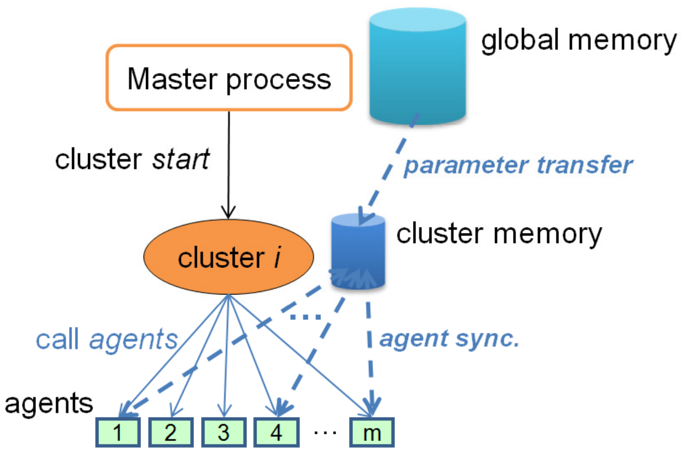


A. Riener, JKU Linz

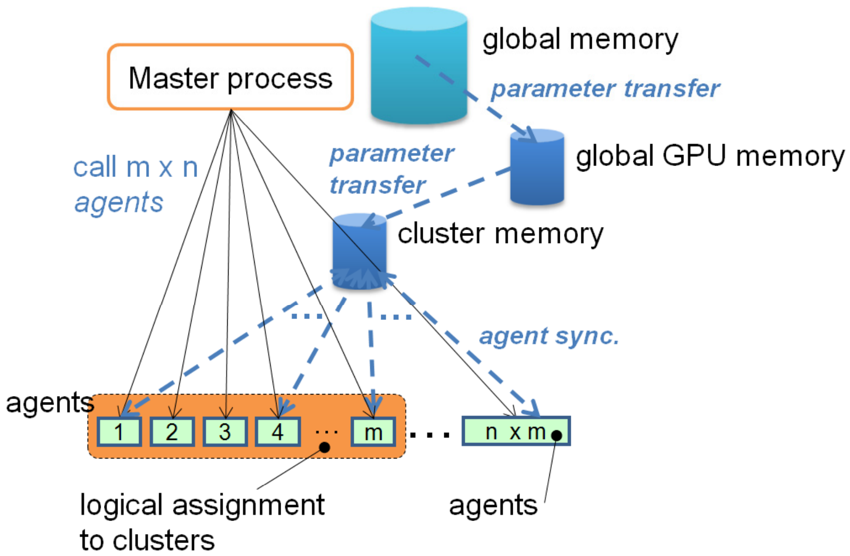
# Parallel, Distributed Simulation (PDS): **Agent Behavior Modeling**

## Simulation setup and synchronization

- **Unique ID** per agent (used for agent communication and data access)
- **No direct** data **exchange** between agents
  - within a cluster via intermediate cluster memory and under control of the cluster thread (for both SMA and GPU)
- Shared memory architecture (**SMA**)



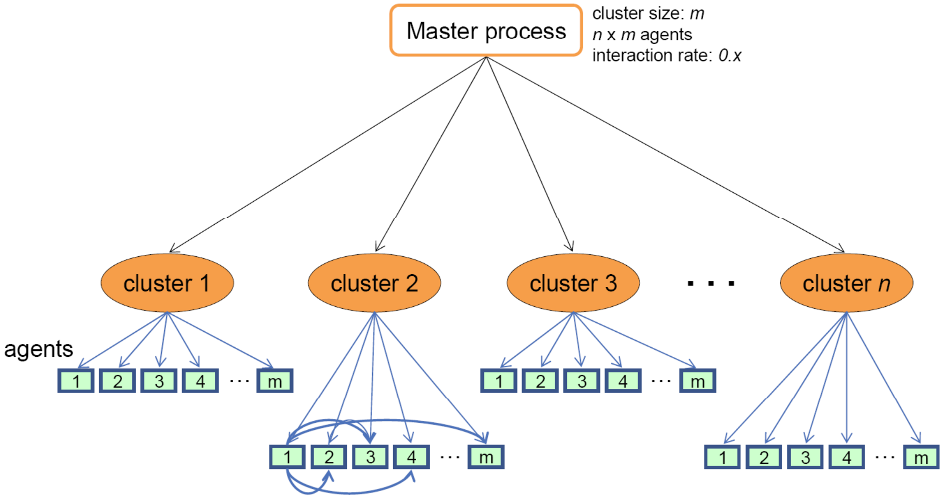

**GPU**



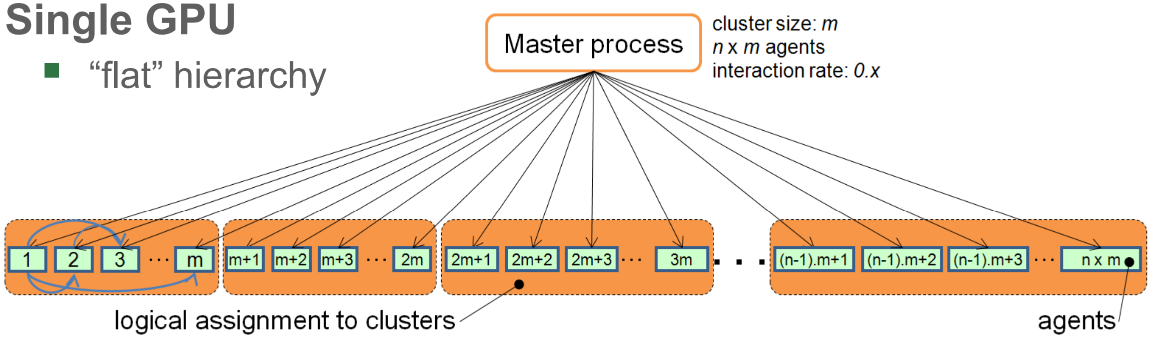
# Parallel, Distributed Simulation (PDS): **Agent Behavior Modeling**

## Simulation execution

- **Shared memory** architecture



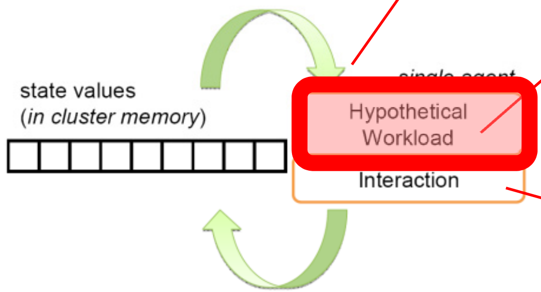

- **Single GPU**
  - “flat” hierarchy



## Parallel, Distributed Simulation (PDS): **Agent Behavior Granularity**

### 1) Cluster behavior model

- **Abstract model** to investigate the influence of the interaction extent/ communication behavior and cluster dimensions on overall performance (agent's **state values** stored in cluster memory)
- **Hypothetic workload** → represents workload a complex model (movement, cognitive adaptation) would generate




- **Interaction** → agent changes its state influenced by agents within interaction range

```
for each agent a in simulation
    stats = synchronize(a);
```

Do something

```
if AmI-assisted agent update-intentions
else update-proximity-parameters

for each exit e
dist = decision-param (e); distance/belief etc.
hope_a(e)= … ;based on group emotions, individualism
fear_a(e)= … ;based on group emotions, individualism
attract_a(e)= …;based on group emotions, individual…

curr-exit = choose exit with max attract
curr-dir = get-direction_curr-exit ;floor field
heading = curr-dir ; setting heading of the agent
move

update-proximity-parameters
For each agent n in neighborhood of a
for each exit e
update group fear, hope and attract
```

Do interact

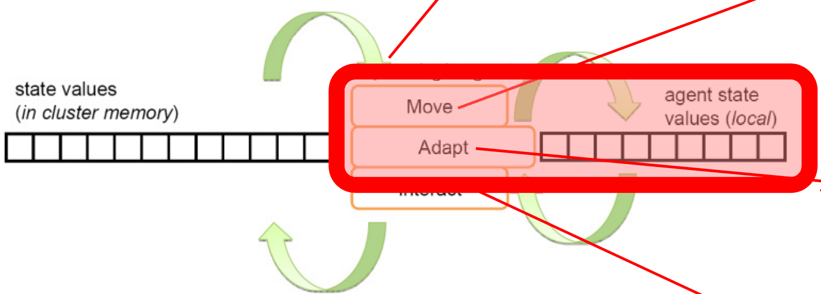
```
update-intentions
update-proximity-parameters
For each agent n in the neighborhood of a
belief_n(curr-exit)= … ;based on belief/trust of group
trust_n(a)= … ;based on belief of a on curr-exit
```


A. Riener, JKU Linz

# Parallel, Distributed Simulation (PDS): **Agent Behavior Granularity**

## 2) Individual behavior model (add *realistic move, adapt procedures*)

- Extension of the **cluster behavior model**
- **Move** → agent changes its position
- **Adapt** → regarding the change in the position, the agent changes its own state values
- **Interact** → agent changes states influenced from different agents




```
for each agent a in simulation
    stats = synchronize(a);
```

```
if AmI-assisted agent update-intentions
else update-proximity-parameters

for each exit e
dist = decision-param (e); distance/belief etc.
hope_a(e)= … ;based on [illegible] emotions, individualism
fear_a(e)= … ;[illegible] group [illegible] individualism
attract_a(e)= … ;[illegible] group emotions, individual…

curr-exit = choose exit with max attract
curr-dir = get-direction_curr-exit ;floor field
heading = curr-dir ; setting heading of the agent
move
```

Perform movement; direction taken from floor field

```
update-proximity-parameters
For each agent n in neighborhood of a
  for each exit e
  update group fear, hope and attract
```

Iterate through neighborhood (information retrieval)

```
update-intentions
update-proximity-parameters
For each agent n in the neighborhood of a
belief_n(curr-exit)= … ;[illegible] belief/trust of group
trust_n(a)= … ;based on belief of a on curr-exit
```

Iterate through neighborhood (update fields)

## Observation

- Cluster memory (state values) is more efficient for interaction compared to direct (1:1) interaction

## **Evaluation of Simulation Results**

cluster behavior model // hypothetical workload

*shared memory architecture (SMA), 1-128 cores*
*single GPU machine*

# Parallel, Distributed Simulation (PDS): **Performance Evaluation**

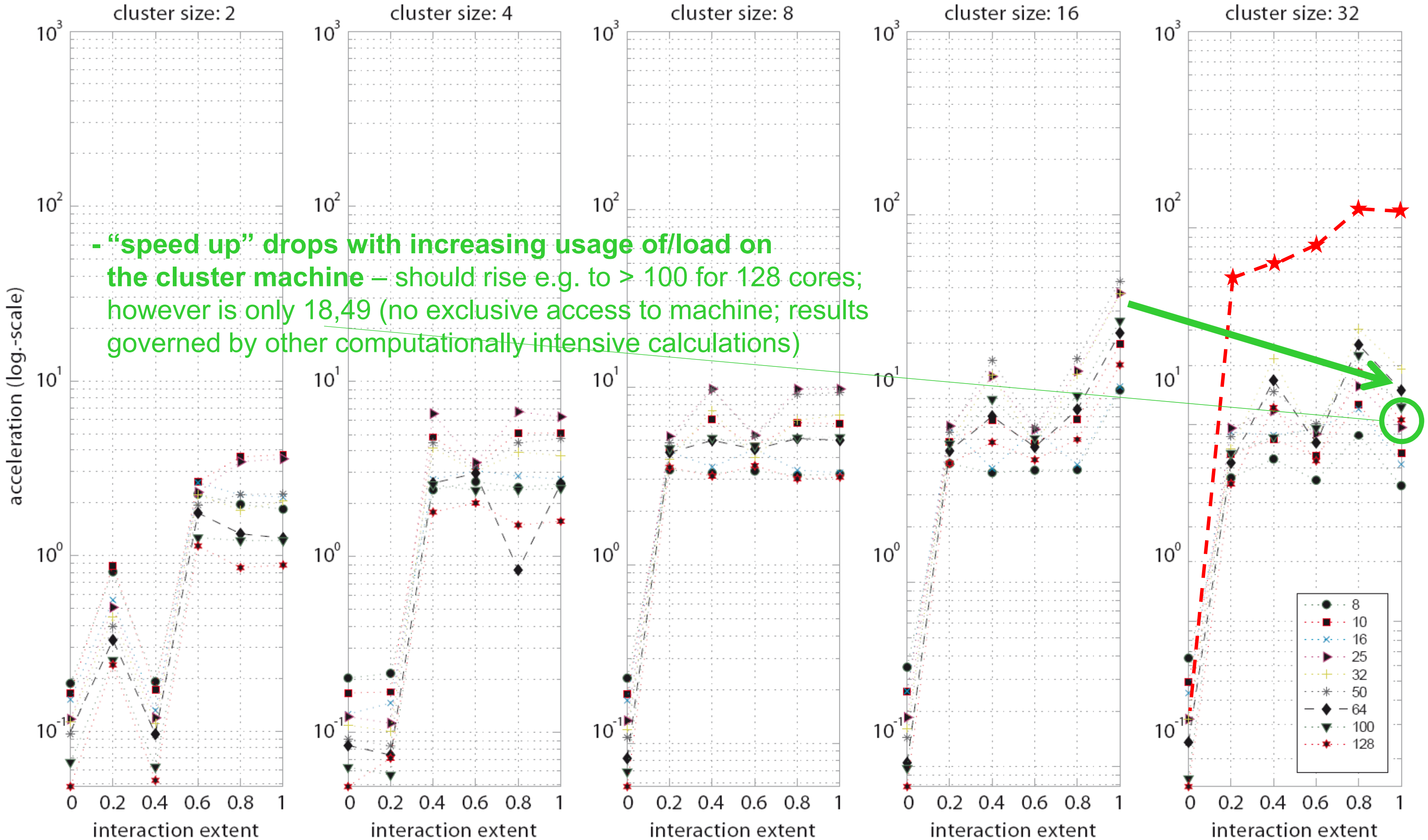
## Evaluation Criteria

- **Model scale:** $10^6$ and $10^7$ agents; distributed at virtual space
- **Cluster size variation:** 2-32 agents/cluster ($2^1$-$2^5$)
- **Connectivity:** interaction between agents within a cluster 0-100% (steps of 20%)
- **Varying number of cores:** 8…128 cores (compared to single core execution) > and related to model execution on one GPU

- **Overall:** 5 (repetitions) x 5 (cluster size) x 6 (connectivity) x 9 (diff. cores) = **1,350 runs** (SMA only)

# Parallel, Distributed Simulation (PDS): **Performance Evaluation**

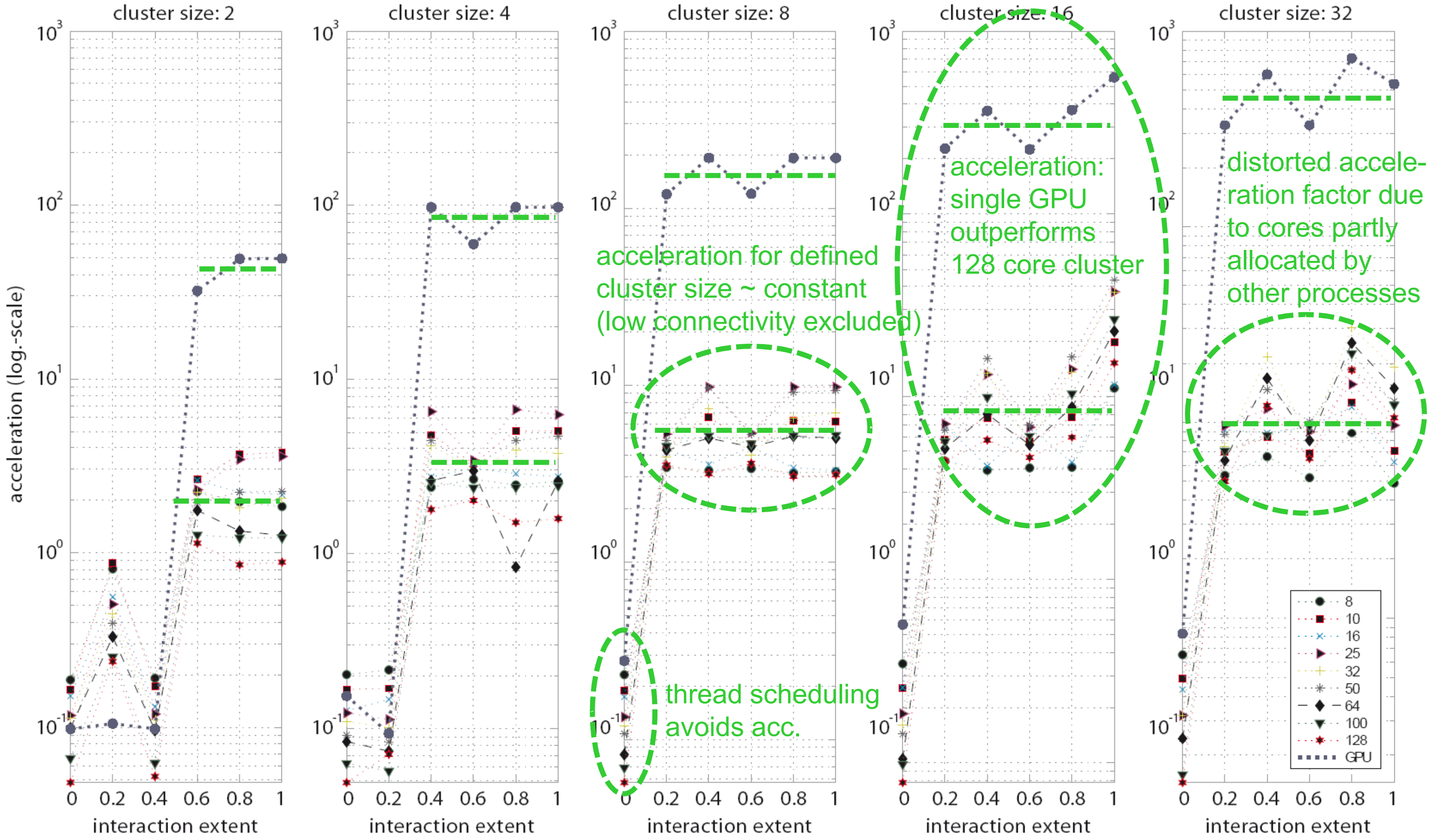
## A) Cluster behavior model (hypothetic workload, $10^7$ agents)




- **"speed up" drops with increasing usage of/load on the cluster machine** – should rise e.g. to > 100 for 128 cores; however is only 18,49 (no exclusive access to machine; results governed by other computationally intensive calculations)

# Parallel, Distributed Simulation (PDS): **Performance Evaluation**

## A) Cluster behavior model (hypothetic workload, $10^7$ agents) - SUMMARY



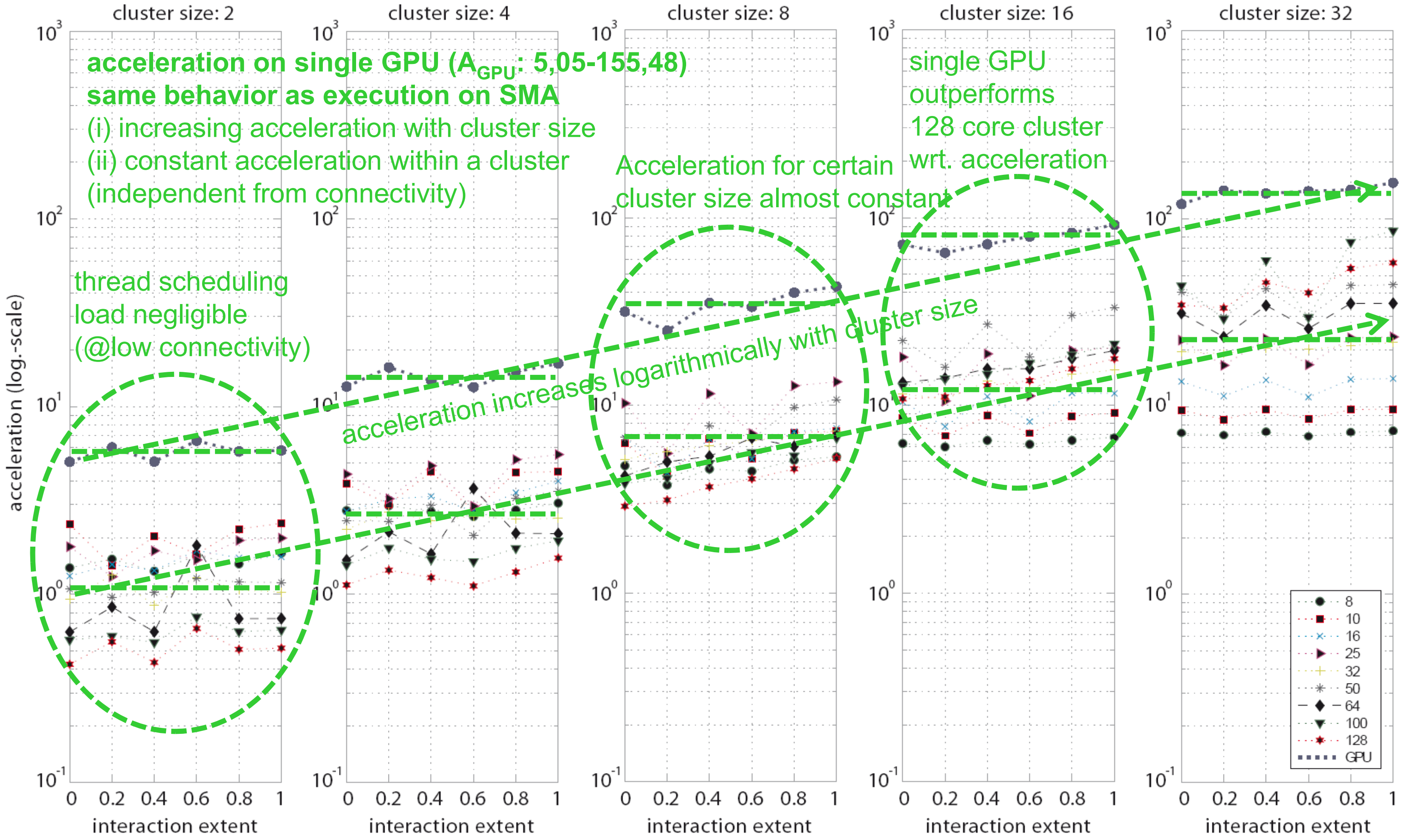
## Evaluation of Simulation Results

individual behavior model // realistic movement, adaptation

*shared memory architecture (SMA), 1-128 cores*
*single GPU machine*

# Parallel, Distributed Simulation (PDS): **Performance Evaluation**

## B) Individual behavior model (“realistic” movement, adaptation, $10^7$) - SUMMARY



# Parallel, Distributed Simulation (PDS): **Performance Evaluation**

## C) Validation of results – acceleration ("relative speed-up")

- Speed-up in performance on different parallel hardware architectures as compared to a atomic (single) processing unit

| Aspect | Facet | Potential Accel. | Description |
|---|---|---|---|
| Individual | Next Step Decision | 90-250 | Computations of the decision for each agent independently [1] |
| Individual | Cognitive Models based on neighborhood | 120 | Optimization of the neighborhood search algorithm [2]** |
| Social | Group belief/trust analysis | 60-80 | Asynchronous read and write operation for uninterrupted computational performance of agents[4]*** |
| Population | Clog detection | 40-90 | Multiple independent search threads launched for detection during the agents are communicating and synchronizing [3]**** |
| Dispersion | Information spread of devices | 50-75 | System wide independent mailbox system for synchronization and communication [3]**** |
| Space/Env. | Implementation of Space | 0 | Same environment representation, but at least 50 times faster during the initialization |
| Mobility | Exit choice Decision | 80-120 | Depending on the Strategy used for making a decision, the data independency must be taken into account [1, 5]* |

*[1] Paul Richmond et. al., A High Performance Agent Based Modelling Framework on Graphics Card Hardware with CUDA, AAMAS 2009, pp. 2.*
*[2] Vincent Garcia et al. , Fast k nearest neighbor search using GPU.*
*[3] Mishra, S.; et al. Parallel and Distributed Systems, IEEE Transactions on Interagent communication and synchronization support in the DaAgent mobile agent-based computing system.*
*[4] Andrew B. Hastings et al., Exploiting shared memory to improve parallel i/o performance, EuroPVMPI08.*
*[5] Christophe Deissenberg et al., EURACE: A Massively Parallel Agent-Based Model of the European Economy, Elsevier, Feb. 2008.*

-----------

*AMD Athlon 2.51 GHz Dual Core Processor with 3GB of RAM and a GeForce 9800 GX2; **Pentium 4 3.4 GHz with 2GB of DDR memory vs. NVIDIA GeForce 8800 GTX; ***Sun FireTM 6800;24 processors at 1.2 GHz and 96 GBytes of RAM;4 Sun StorEdgeTM T3;1 Gbit Fibrechannel;Sun StorageTekTM QFS 4.5 filesystem; Net. of diff. OS machines

# Parallel, Distributed Simulation (PDS): **Performance Evaluation**

## Conclusive remarks

- **Execution time** (i.e., acceleration) scales
  - **almost linear** with no. of agents ($10^6 \rightarrow 10^7$; not shown)
  - **logarithmically** with the cluster size (tested from $2^1$ to $2^5$)

- **GPU outperforms SMA** (cluster machine) – even with 100+ cores
  - cluster behavior model (coarse-grained): 43.50 vs. 536.66
  - individual behavior model (fine-grained): 86.47 vs. 155.48

  → increasing computation in individual agents reduces the gain in acceleration

- **Connectivity** (i.e., rate of interaction) between agents in a cluster **does not influence acceleration** much…
  - Except for the coarse-grained model and low/no interaction: scheduling, thread switching prevents from acceleration with more cores…

## Parallel, Distributed Simulation (PDS): **Performance Evaluation**

### Options to further improve execution performance

- **Parallel execution** (either on a GPU or cluster) is used to fulfill the need of high computational power for large-scale ABM's

- To further **accelerate simulation**, computational and communicational methods should be distinguished
  - computational methods: executed without any interaction with the simulation environment (space, agents)
  - communicational methods: computing their results based on observations from the vicinity and, in case of AmI support, information spread of the AmI device

- **Computational methods** can be arranged to fill **CPU idle** periods
  → performance increase on accordingly rearranged threads

D. Moser, A. Riener, K. Zia, A. Ferscha
*Department for Pervasive Computing, JKU Linz/Austria*

# “Parallel Simulation of Social Agents using Cilk and OpenCL”

**DS-RT 2011**
15th International Symposium on Distributed Simulation and Real Time Applications
September 4,-7, 2011, Salford/Manchester, UK

This work is supported under the FP7 ICT FET program of the European Commission under grant agreement No 231288 (SOCIONICAL)

**Dr. Andreas Riener**
JKU Linz, Department for Pervasive Computing
Altenberger Straße 69, A-4040 Linz
www.pervasive.jku.at/about_us/staff/riener