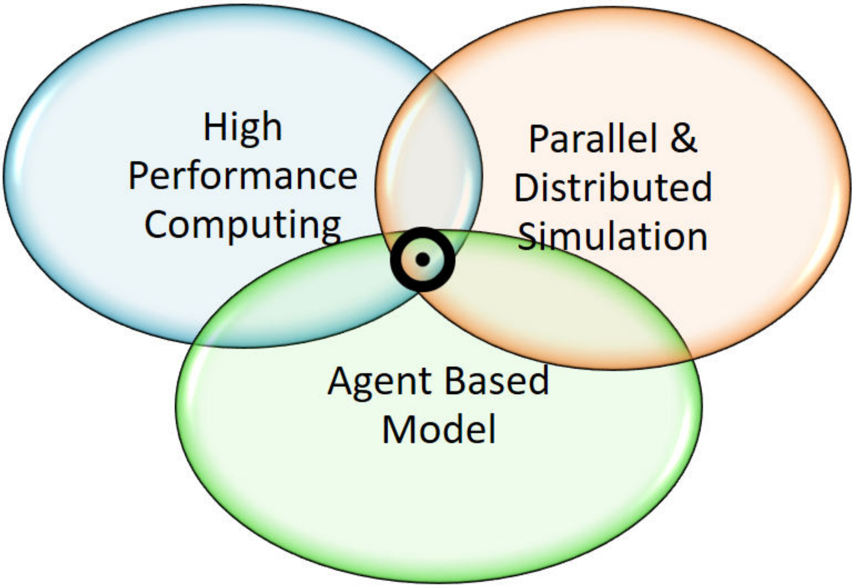
Care HPS: A High Performance Simulation Methodology for Complex Agent-Based Models

Francisco Borges

Advisor: Dr. Remo Suppi

Federal Institute of Bahia
and
Universitat Autònoma de Barcelona
Computer Architecture & Operating Systems Department

February 21, 2017, Salvador


High
Performance
Computing
Parallel &
Distributed
Simulation
Agent Based
Model

What is agent-based model?

- A fixed number of discrete and autonomous agents.
- Local rules are applied to each agent.
- An environment.
- Collective behaviors emerge as a result of local interaction between agents.

What is agent-based model?

- A fixed number of discrete and autonomous agents.
- Local rules are applied to each agent.
- An environment.
- Collective behaviors emerge as a result of local interaction between agents.

What is agent-based model?

- A fixed number of discrete and autonomous agents.
- Local rules are applied to each agent.
- An environment.
- Collective behaviors emerge as a result of local interaction between agents.

What is agent-based model?

- A fixed number of discrete and autonomous agents.
- Local rules are applied to each agent.
- An environment.
- Collective behaviors emerge as a result of local interaction between agents.

As an example: fish schooling

Requirements for simulating complex system using ABMS

- Simulations must offer realistic results.
- It means simulations whose results are valid in reality, and which can also be used for prediction or to explain some phenomenon.
- Therefore, these simulations **require reliable** results through statistical approaches.
- Moreover, they have a high computational complexity because thousands of agents model them, their higher level of parameters and their behavior complexity. Thus, this kind of simulation **requires a long execution time**.

Requirements for simulating complex system using ABMS

- Simulations must offer realistic results.
- It means simulations whose results are valid in reality, and which can also be used for prediction or to explain some phenomenon.
- Therefore, these simulations **require reliable** results through statistical approaches.
- Moreover, they have a high computational complexity because thousands of agents model them, their higher level of parameters and their behavior complexity. Thus, this kind of simulation **requires a long execution time**.

Requirements for simulating complex system using ABMS

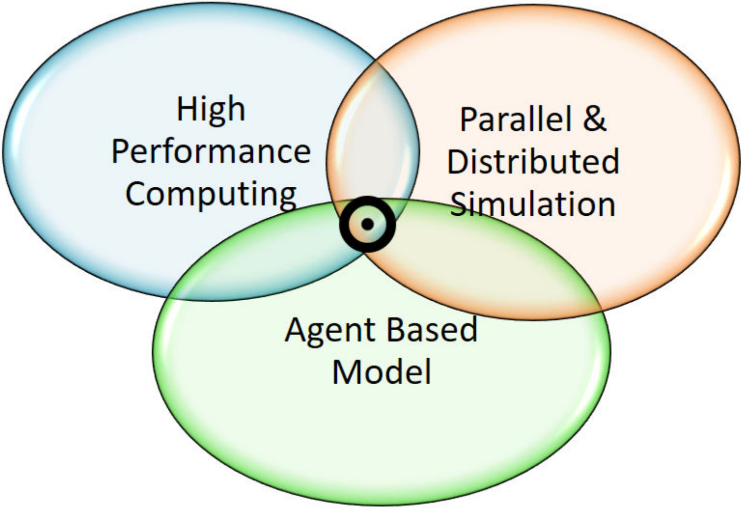
- Simulations must offer realistic results.
- It means simulations whose results are valid in reality, and which can also be used for prediction or to explain some phenomenon.
- Therefore, these simulations **require reliable** results through statistical approaches.
- Moreover, they have a high computational complexity because thousands of agents model them, their higher level of parameters and their behavior complexity. Thus, this kind of simulation **requires a long execution time**.

Requirements for simulating complex system using ABMS

- Simulations must offer realistic results.
- It means simulations whose results are valid in reality, and which can also be used for prediction or to explain some phenomenon.
- Therefore, these simulations **require reliable** results through statistical approaches.
- Moreover, they have a high computational complexity because thousands of agents model them, their higher level of parameters and their behavior complexity. Thus, this kind of simulation **requires a long execution time**.

We take HPC techniques for solving real problems. These problems are complex system modeled with ABM. Its execution using parallel and distributed simulation enable us reach conclusions and gain knowledge.

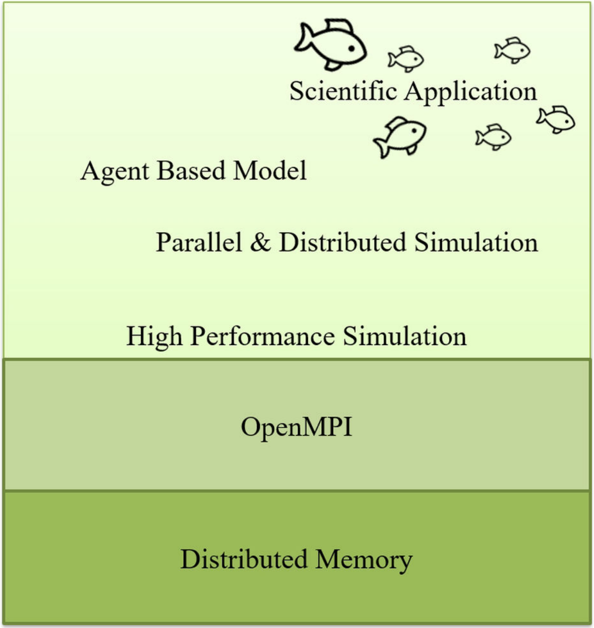
We take HPC techniques for solving real problems. These problems are complex system modeled with ABM. Its execution using parallel and distributed simulation enable us reach conclusions and gain knowledge.

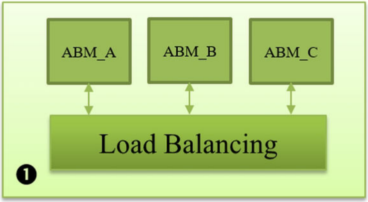


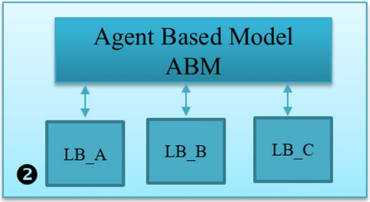
Outline

Outline

1. Introduction
   - Motivation and justification
   - Research question
   - Literature review
   - Objective


Scientific Application
Agent Based Model
Parallel & Distributed Simulation
High Performance Simulation
OpenMPI
Distributed Memory


ABM_A
ABM_B
ABM_C
Load Balancing
❶


Agent Based Model
ABM
LB_A
LB_B
LB_C
❷

- How to generalize our HPC techniques and approaches for agent-based models that demand high performance solutions?

- We can find several ABMS tools.
- A few of them **use HPC** to **execute** agent-based models.
- Although exist many consolidated ABMS tools **none of them** enable the **experimentation** in HPC.

- We can find several ABMS tools.
- A few of them **use HPC** to **execute** agent-based models.
- Although exist many consolidated ABMS tools **none of them** enable the **experimentation** in HPC.

- We can find several ABMS tools.
- A few of them **use HPC** to **execute** agent-based models.
- Although exist many consolidated ABMS tools **none of them** enable the **experimentation** in HPC.

Aims

We introduce a methodology called Care High Performance Simulation (HPS). The main objective of Care HPS is:

- to be a scientific instrument for research on **HPC** for **agent-based models** that **demand high performance solutions**.

Aims

We introduce a methodology called Care High Performance Simulation (HPS). The main objective of Care HPS is:

- to be a scientific instrument for research on **HPC** for **agent-based models** that **demand high performance solutions**. Its sub-objectives are:
  - to enable application area researchers to study, design and implement complex ABM models that require an HPC solution; and,
  - to enable HPC experts to develop techniques and solutions of high performance distributed simulation for agent-based models.

Aims

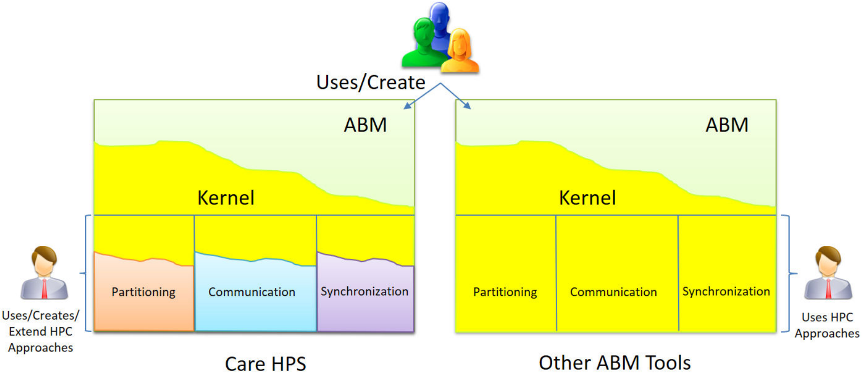
We introduce a methodology called Care High Performance Simulation (HPS). The main objective of Care HPS is:

- to be a scientific instrument for research on **HPC** for **agent-based models** that **demand high performance solutions**. Its sub-objectives are:
  - to enable application area researchers to study, design and implement complex ABM models that require an HPC solution; and,
  - to enable HPC experts to develop techniques and solutions of high performance distributed simulation for agent-based models.

Aims

We introduce a methodology called Care High Performance Simulation (HPS). The main objective of Care HPS is:

- to be a scientific instrument for research on **HPC** for **agent-based models** that **demand high performance solutions**. Its sub-objectives are:
  - to enable application area researchers to study, design and implement complex ABM models that require an HPC solution; and,
  - to enable HPC experts to develop techniques and solutions of high performance distributed simulation for agent-based models.



Uses/Create
ABM
Kernel
Partitioning
Communication
Synchronization
Uses/Creates/
Extend HPC
Approaches
Care HPS
ABM
Kernel
Partitioning
Communication
Synchronization
Uses HPC
Approaches
Other ABM Tools

Contributions

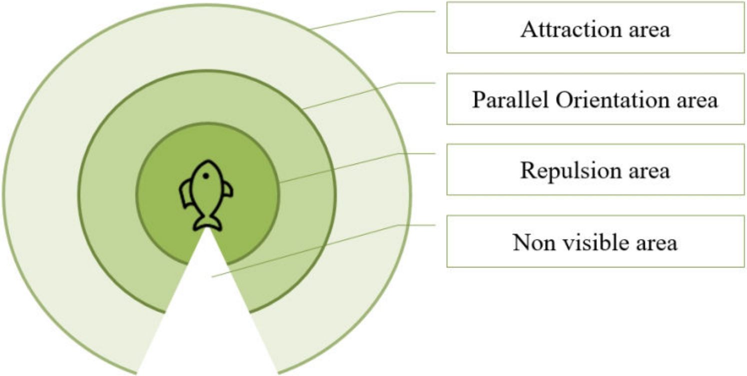
2. ABMs case study
3. HPC approaches
4. Care HPS

Outline

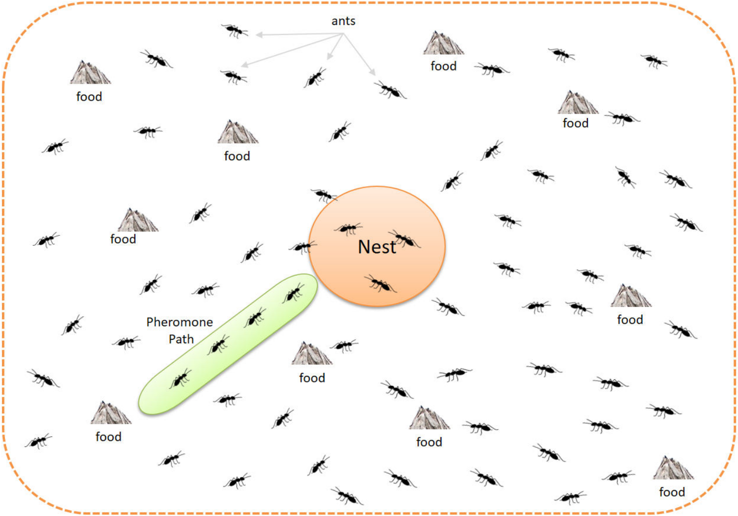
Agent-based models overview

Model	Agents	Type	Real Application?
Fish Schooling	Fish	Biological	No
Ant colony	Ant	Biological	No
Shopping agent	Buyer	Sociological	No
Aedes Aegypti (mosquito)	Mosquito, Persons, Health Agent	Biological	Yes

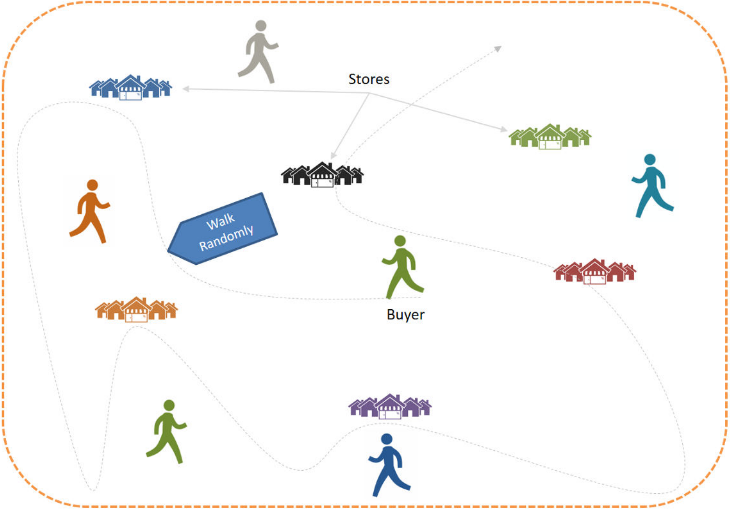
Huth and Wissel model



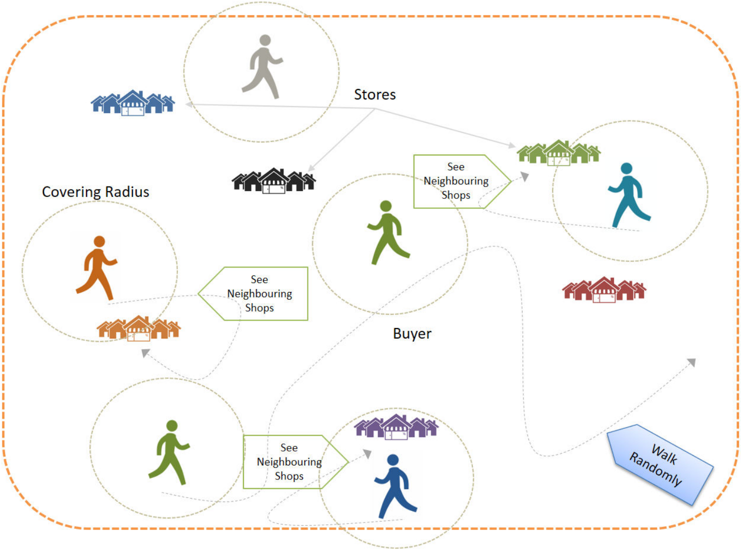
Wilensky model



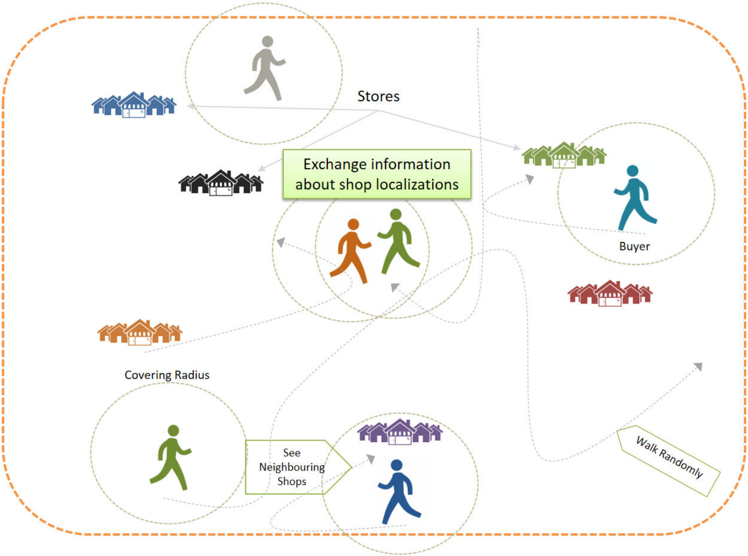
Gilbert and Troitzsch Model - simplest version



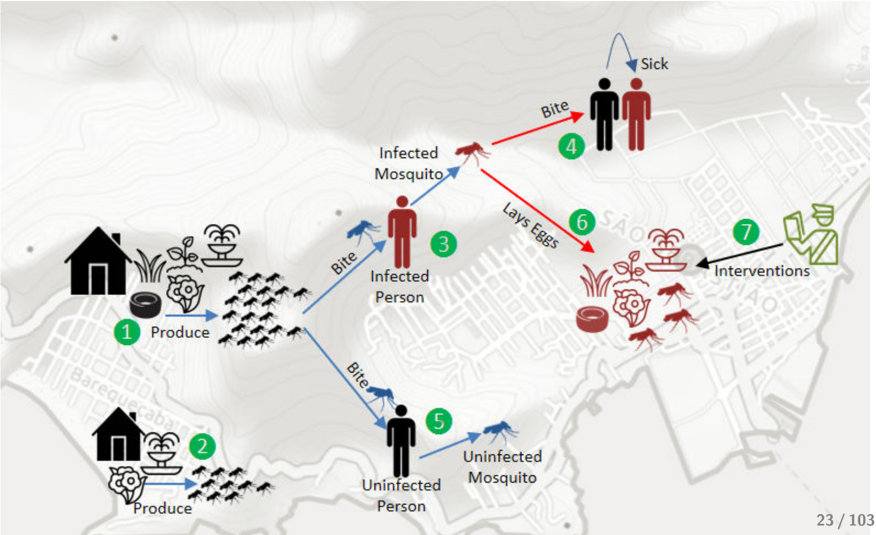
Gilbert and Troitzsch model - add another behavior



Gilbert and Troitzsch model - smartest agents

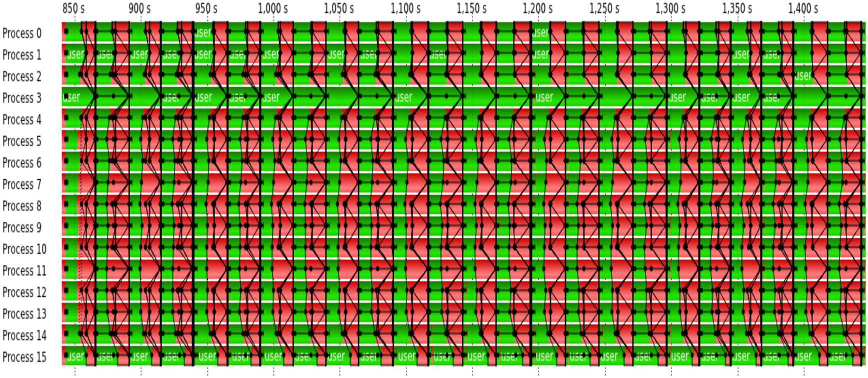


Assessment of Aedes Aegypti pupal productivity model



Outline

HPC approaches: Communication patterns



Goals

Goals

Primary goals

We have compared three communication strategies:

- Asynchronous and synchronous message passing.
- Bulk-synchronous parallel (BSP) for communication.

Goals

Primary goals

We have compared three communication strategies:

- Asynchronous and synchronous message passing.
- Bulk-synchronous parallel (BSP) for communication.

Goals

Primary goals

We have compared three communication strategies:

- Asynchronous and synchronous message passing.
- Bulk-synchronous parallel (BSP) for communication.

Goals

Primary goals

We have compared three communication strategies:

- Asynchronous and synchronous message passing.
- Bulk-synchronous parallel (BSP) for communication.

Why?

- As said before, parallel and distributed simulation has problems with communication.
- Therefore, we implemented different methods of communication to evaluate the scalability.

Goals

Primary goals

We have compared three communication strategies:

- Asynchronous and synchronous message passing.
- Bulk-synchronous parallel (BSP) for communication.

Why?

- As said before, parallel and distributed simulation has problems with communication.
- Therefore, we implemented different methods of communication to evaluate the scalability.

Goals

Primary goals

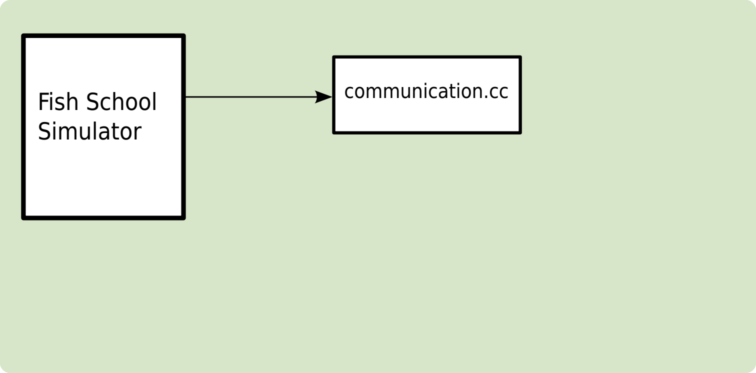
We have compared three communication strategies:

- Asynchronous and synchronous message passing.
- Bulk-synchronous parallel (BSP) for communication.

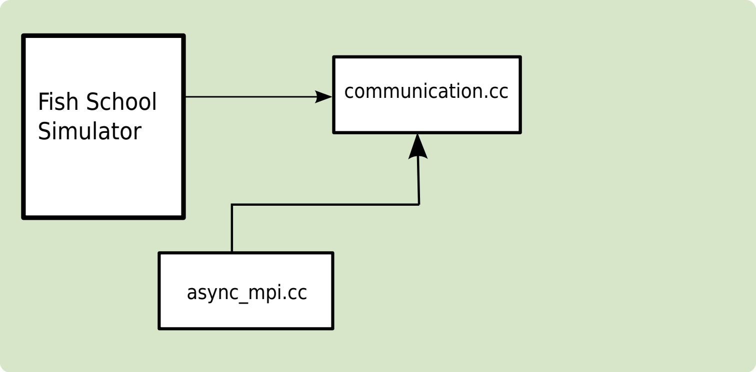
Why?

- As said before, parallel and distributed simulation has problems with communication.
- Therefore, we implemented different methods of communication to evaluate the scalability.

Implementation



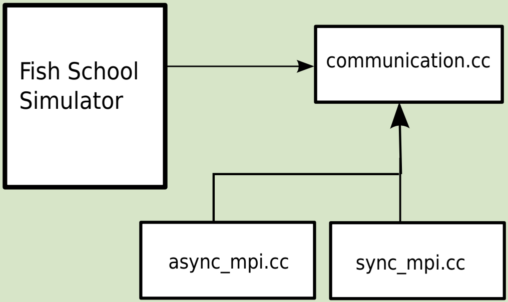
Implementation



Message passing via MPI

- MPI_Isend
- MPI_Irecv

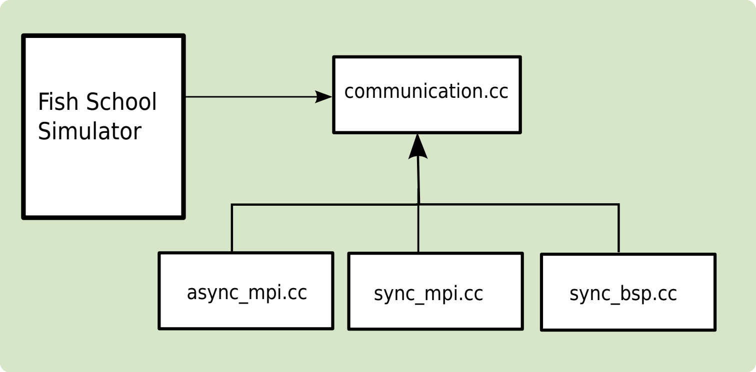
Implementation



Message passing via MPI

- MPI_Send
- MPI_Recv

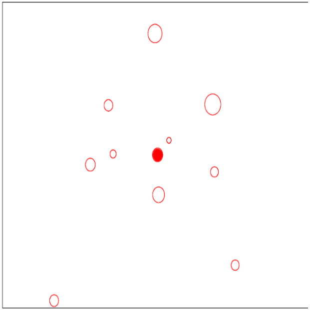
Implementation



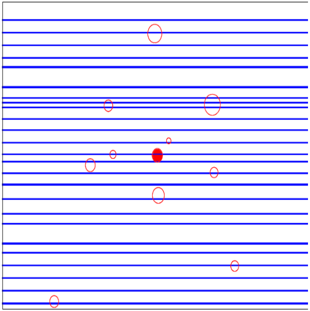
Bulk-synchronous parallel via BSPonMPI

- BSPonMPI is a small communications library for BSP which consists of only 20 basic operations on top of MPI.

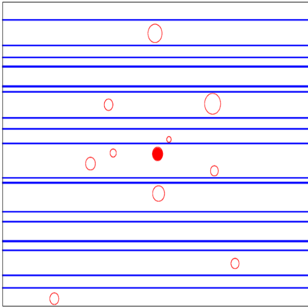
HPC approaches: Strip partitioning



Ant colony environment

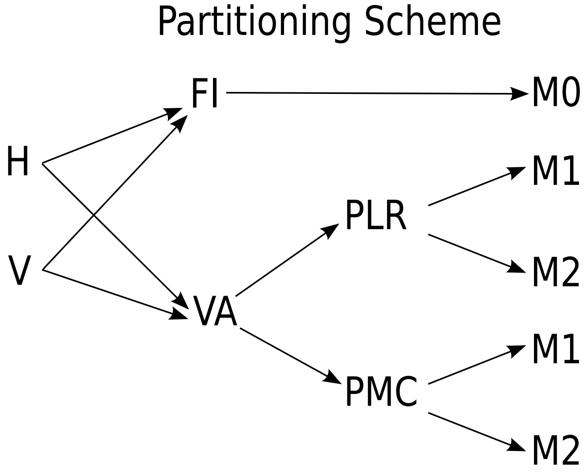


Environment partitioned with sharing objects



Environment partitioned with no sharing objects

Strip partitioning



HPC approaches: Hybrid strip partitioning approach

- We extended the strip partitioning algorithm.
- We decrease the idleness of these cores through the creation of OpenMP threads which are used to compute the extra agents that are in other cores.
- This partitioning checks the proportion of the quantity of agents inside a strip and dynamically creates a number of threads.

HPC approaches: Hybrid strip partitioning approach

- We extended the strip partitioning algorithm.
- We decrease the idleness of these cores through the creation of OpenMP threads which are used to compute the extra agents that are in other cores.
- This partitioning checks the proportion of the quantity of agents inside a strip and dynamically creates a number of threads.

HPC approaches: Hybrid strip partitioning approach

- We extended the strip partitioning algorithm.
- We decrease the idleness of these cores through the creation of OpenMP threads which are used to compute the extra agents that are in other cores.
- This partitioning checks the proportion of the quantity of agents inside a strip and dynamically creates a number of threads.

Motivation

HPC approaches: Optimal run length for simulations

- In simulations the results usually come from a stochastic process.
- How to compare these solutions since the results are not deterministic?
- Consequently how to guarantee that the output results are statistically trusted?
- We apply a statistical approach in order to define the optimal run length for simulations.

Motivation

HPC approaches: Optimal run length for simulations

- In simulations the results usually come from a stochastic process.
- How to compare these solutions since the results are not deterministic?
- Consequently how to guarantee that the output results are statistically trusted?
- We apply a statistical approach in order to define the optimal run length for simulations.

Motivation

HPC approaches: Optimal run length for simulations

- In simulations the results usually come from a stochastic process.
- How to compare these solutions since the results are not deterministic?
- Consequently how to guarantee that the output results are statistically trusted?
- We apply a statistical approach in order to define the optimal run length for simulations.

Motivation

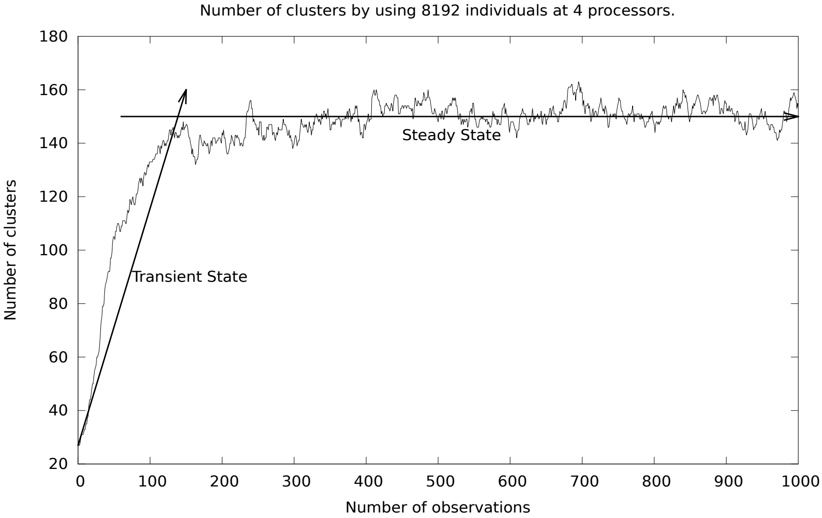
HPC approaches: Optimal run length for simulations

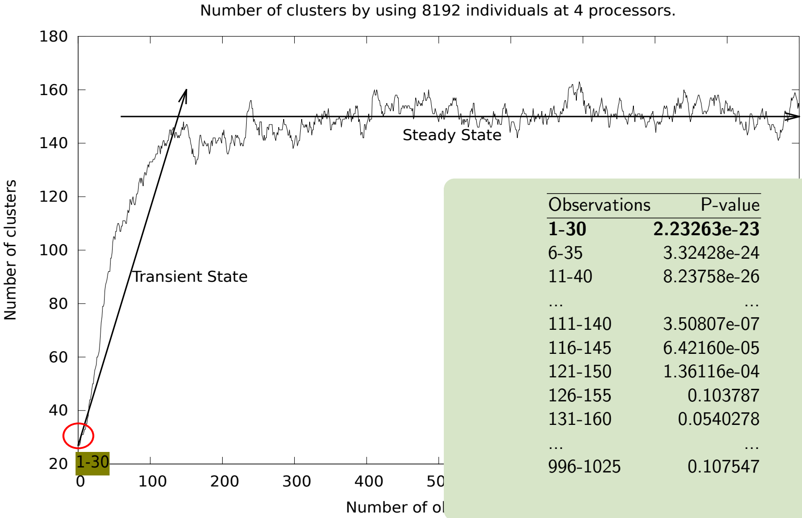
- In simulations the results usually come from a stochastic process.
- How to compare these solutions since the results are not deterministic?
- Consequently how to guarantee that the output results are statistically trusted?
- We apply a statistical approach in order to define the optimal run length for simulations.

Method steps

1. Identify the steady state
2. Identify the run length
   - Identify the non significant correlation lag size
   - Make a batch ten times the size of the lag
   - Make the steady state replication run length ten batches long
3. Replication analysis
   - Ensure that the number of replication is enough
   - Determine if the means are statistically significantly different from the others
   - Identify which means are different

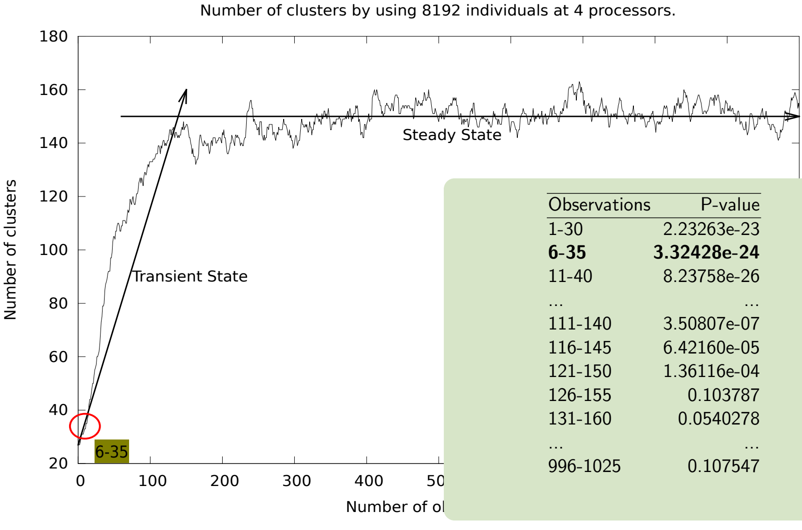
Optimal run length for simulations



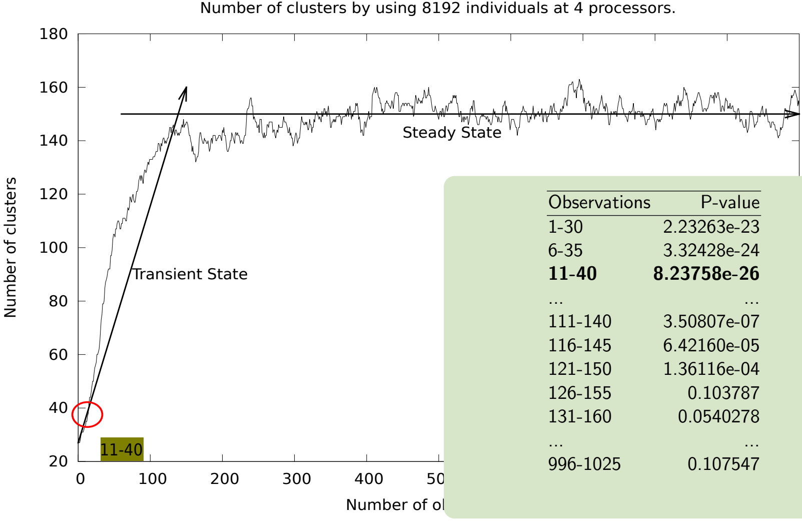


Observations	P-value
1-30	**2.23263e-23**
6-35	3.32428e-24
11-40	8.23758e-26
...	...
111-140	3.50807e-07
116-145	6.42160e-05
121-150	1.36116e-04
126-155	0.103787
131-160	0.0540278
...	...
996-1025	0.107547

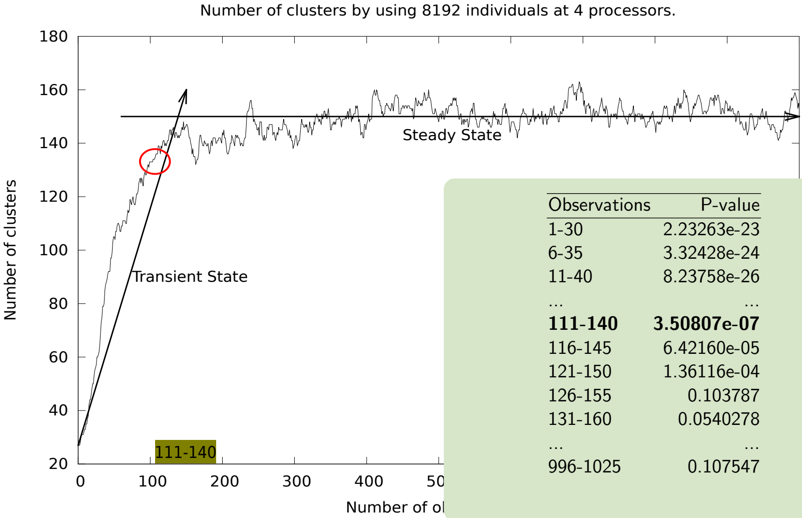
Optimal run length for simulations



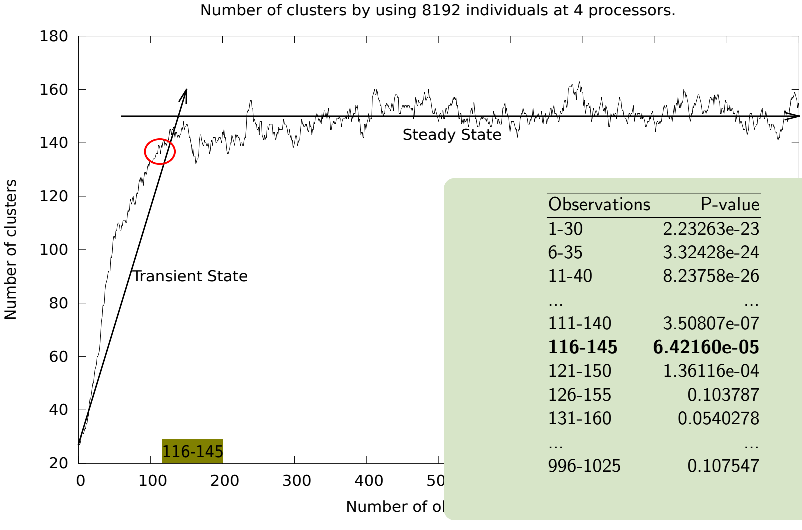
Observations	P-value
1-30	2.23263e-23
6-35	**3.32428e-24**
11-40	8.23758e-26
...	...
111-140	3.50807e-07
116-145	6.42160e-05
121-150	1.36116e-04
126-155	0.103787
131-160	0.0540278
...	...
996-1025	0.107547



Observations	P-value
1-30	2.23263e-23
6-35	3.32428e-24
11-40	**8.23758e-26**
...	...
111-140	3.50807e-07
116-145	6.42160e-05
121-150	1.36116e-04
126-155	0.103787
131-160	0.0540278
...	...
996-1025	0.107547

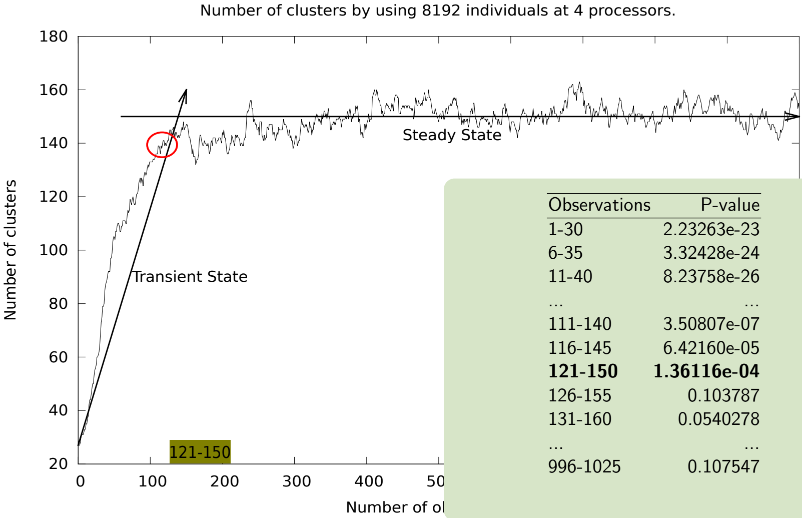


Observations	P-value
1-30	2.23263e-23
6-35	3.32428e-24
11-40	8.23758e-26
...	...
111-140	**3.50807e-07**
116-145	6.42160e-05
121-150	1.36116e-04
126-155	0.103787
131-160	0.0540278
...	...
996-1025	0.107547



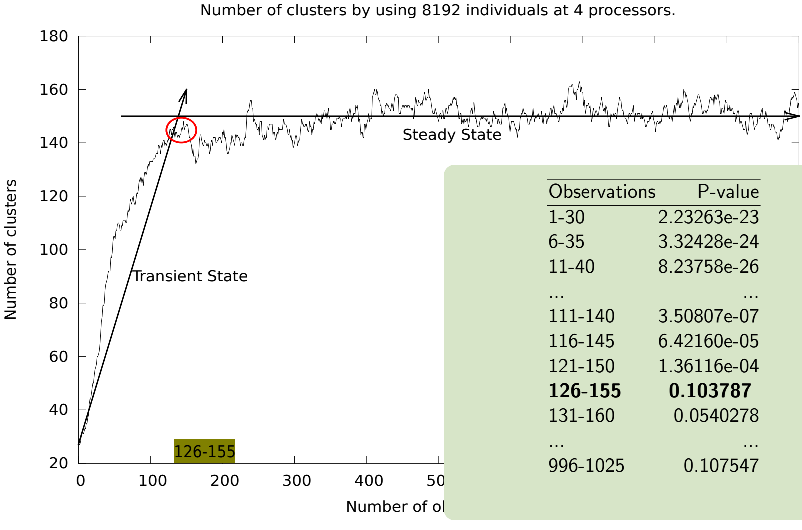
Observations	P-value
1-30	2.23263e-23
6-35	3.32428e-24
11-40	8.23758e-26
...	...
111-140	3.50807e-07
116-145	**6.42160e-05**
121-150	1.36116e-04
126-155	0.103787
131-160	0.0540278
...	...
996-1025	0.107547

Optimal run length for simulations



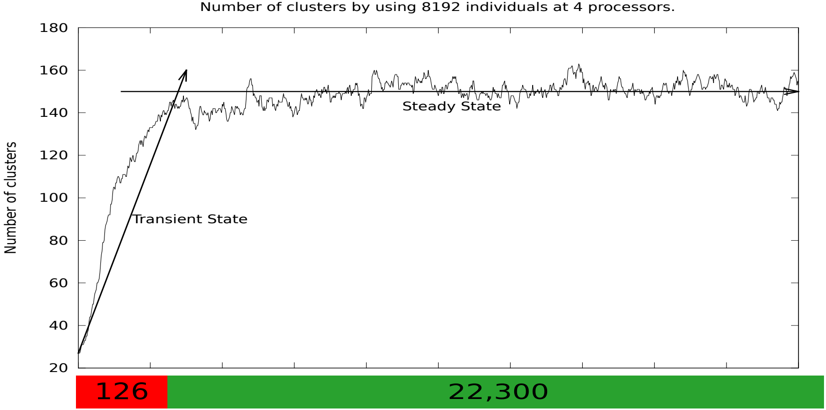
Observations	P-value
1-30	2.23263e-23
6-35	3.32428e-24
11-40	8.23758e-26
...	...
111-140	3.50807e-07
116-145	6.42160e-05
121-150	**1.36116e-04**
126-155	0.103787
131-160	0.0540278
...	...
996-1025	0.107547

Optimal run length for simulations



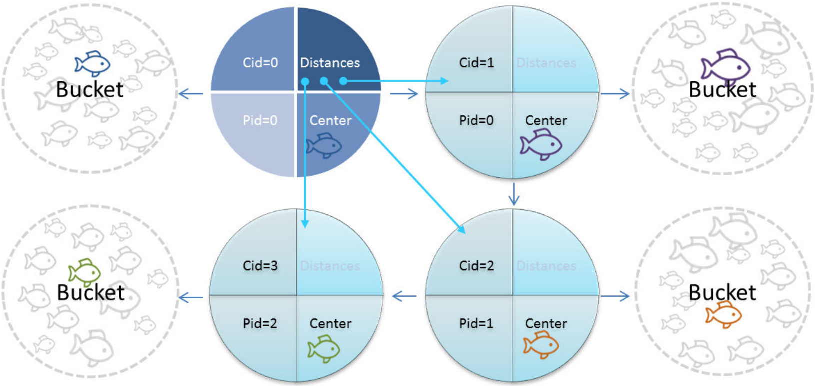
Observations	P-value
1-30	2.23263e-23
6-35	3.32428e-24
11-40	8.23758e-26
...	...
111-140	3.50807e-07
116-145	6.42160e-05
121-150	1.36116e-04
126-155	**0.103787**
131-160	0.0540278
...	...
996-1025	0.107547

Optimal run length for simulations

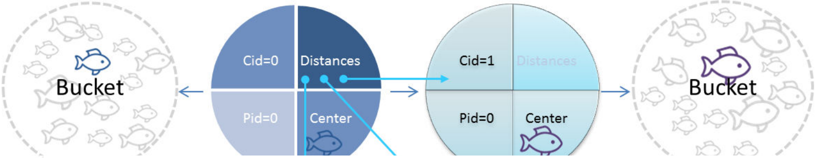


Lag size = 223
Batch size = 2,230
Total number of observations = 22,426

Distributed memory data structure

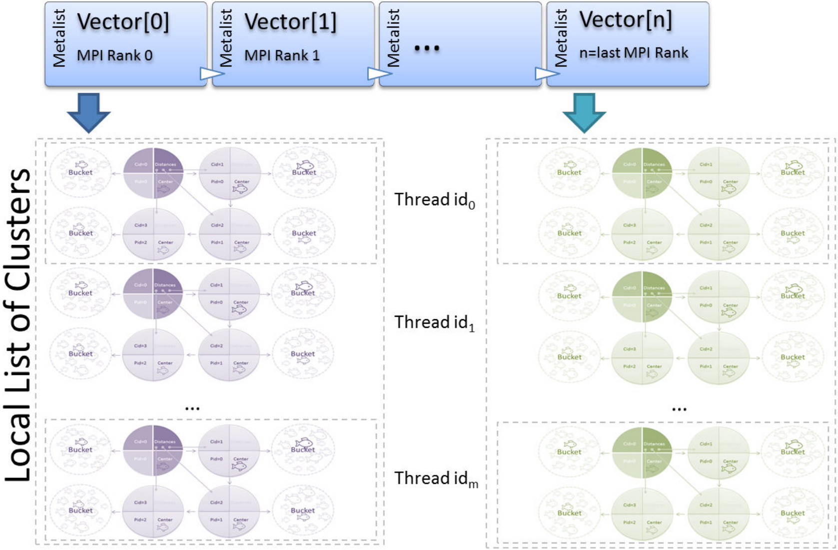


Distributed memory data structure



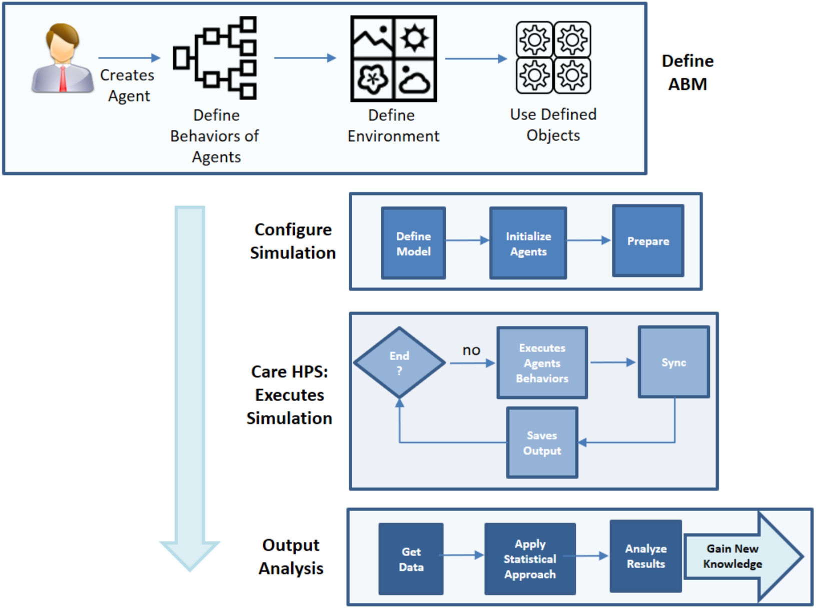

```
#ifdef OPENMP
#pragma omp critical (neighbors_exchange_lc_send)
                                  {
#endif
                                       index = lc_send[pid]->length();
                                       lc_send[pid]->insert(c);
#ifdef OPENMP
                                  }
#endif
```

Shared memory data structure

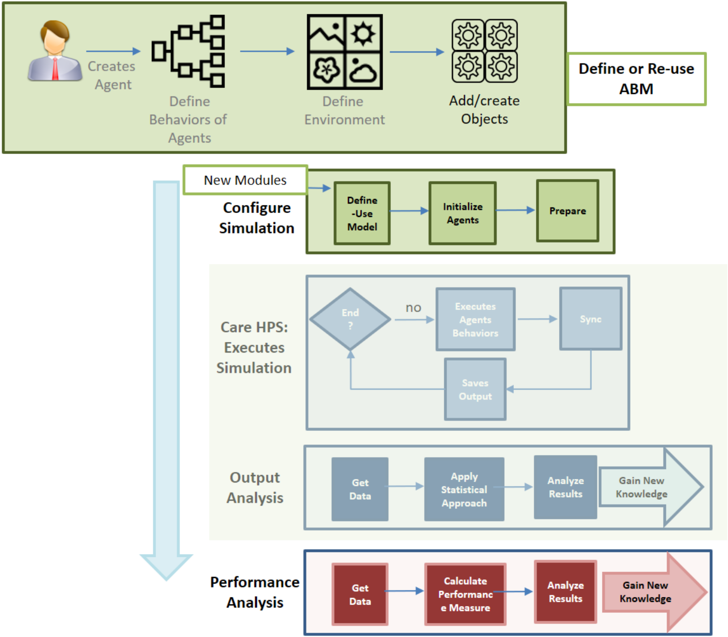


Outline

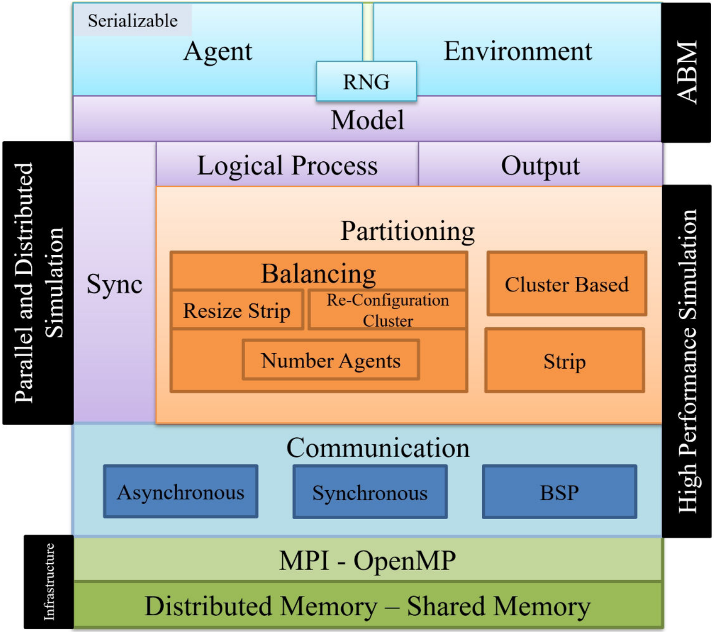
Methodology for application area user



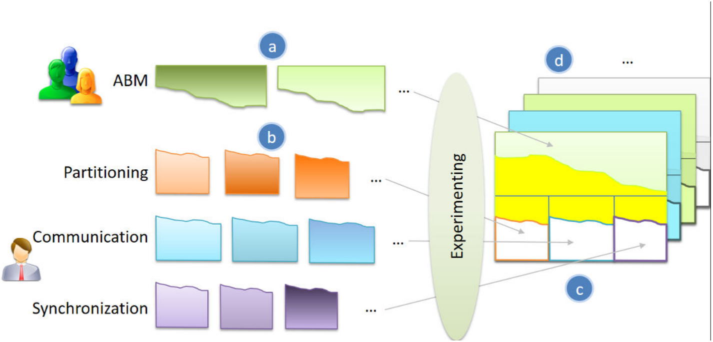
Methodology for HPC expert



Care HPS architecture



Several scenarios can be represented



Design and programming issues

Object-oriented design

- Design pattern

Design and programming issues

Object-oriented design

- Design pattern

Design and programming issues

Object-oriented design

- Design pattern

Object-oriented programming

- Inheritance
- Polymorphism
- Interface

Design and programming issues

Object-oriented design

- Design pattern

Object-oriented programming

- Inheritance
- Polymorphism
- Interface

Design and programming issues

Object-oriented design

- Design pattern

Object-oriented programming

- Inheritance
- Polymorphism
- Interface

Design and programming issues

Object-oriented design

- Design pattern

Object-oriented programming

- Inheritance
- Polymorphism
- Interface

Creating an environment for the fish

```
pds_fish* PDS_FISH=NULL;
ABM_fish* ABM=NULL;
environment* ENV=NULL;
PDS_FISH = new pds_fish();
ABM = PDS_FISH->createABM();

/** Creates an environment with a
plan(ax + by + cz + d = 0) defined by
equation: 2*x - 13*y + 5 */

ENV = PDS_FISH → createEnvironment(new linear_plan( 2,-13,0,5));

/** The user can create how many objects inside of
the environment that are required. */
/** Code that creates other objects inside of
the environment defined by the equation: 2*y-3 */

// ENV → createObject(new linear_plan(0,2,0,-3)); //
```

Implements the interaction with environment

```
vector<object*> obj_env;
obj_env = ENV->getObjects();
for (vector<object*>::iterator ob=obj_env.begin();
        ob!=obj_env.end();
        ob++)

 if (  (*ob)→check_collision(this->get_position(),
                             this->get_velocity(),
                             MAXIMUM_VISION_RANGE))
     this→repulsion(*this);
}
```

Creating the partitioning class

```
class partitioning_strip_hybrid : public
        partitioning_strip{

  public :
      // Contructor methods

      // Override the execute method
      void execute(int);


};
```

Defining the partitioning strategy

```
//here goes other the factory methods of the model class.

partitioning* model_ant::factory_partitioning(){
  return new partitioning_strip_hybrid();

}
```

Outcomes

5. Results
6. Publications

Outline

Results: Verification of containers productivity

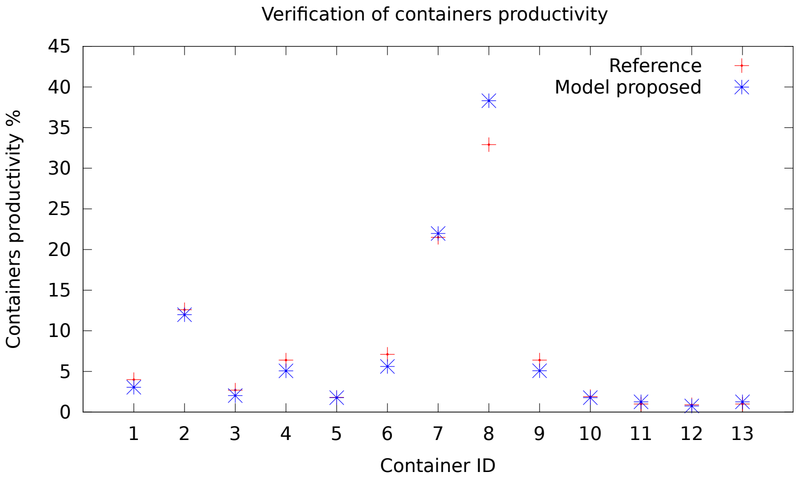
Objective

Comparative between the container pupal productivity with reference to the average of the percentage of pupae per container obtained from the proposed model.

Data and parameters

Number of simulations	1500
Days simulated	100
Measure analyzed	Containers productivity %

Verification of containers productivity



Results: Container pupal productivity

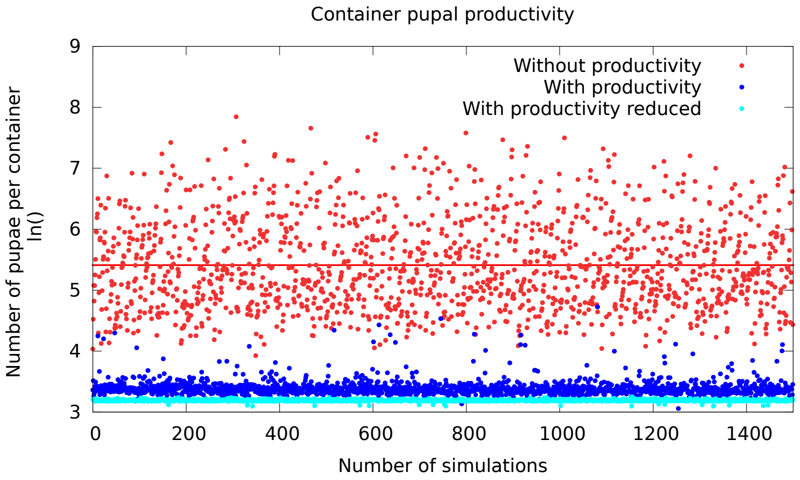
Objective

Present the effects of consider the pupal productivity issue in the number of pupae per container.

Data and parameters

Number of simulations	1500
Days simulated	100
Measure analyzed	Number of pupal per containers ln() scale.

Container pupal productivity



Results: What-if

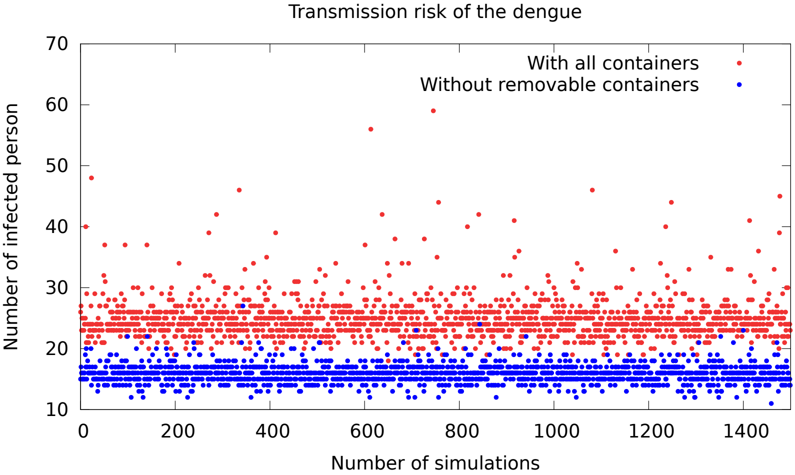
Objective

Hypothetical situation where a health agent changes the model parameters in order to simulate actions for decisions made.

Data and parameters

Number of simulations	1500
Days simulated	100
Measure analyzed	Number of infected person

What-if



Transmission risk of the dengue
With all containers
Without removable containers
Number of infected person
Number of simulations
70
60
50
40
30
20
10
0
200
400
600
800
1000
1200
1400

Results: Area of the mosquito actuation.

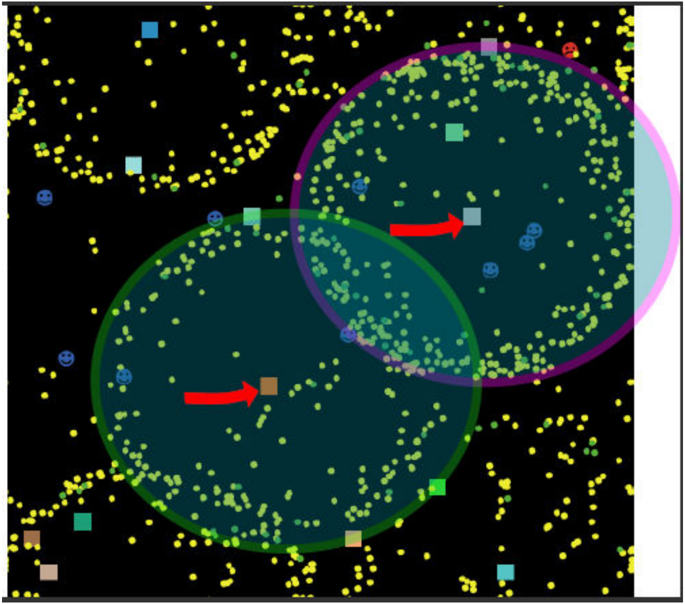
Objective

Emergent behavior of the mosquitoes.

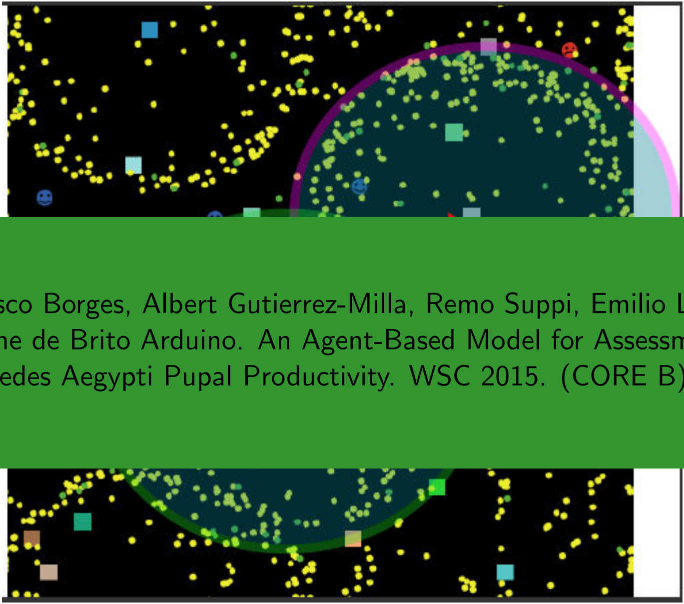
Data and parameters

Number of simulations	1500
Days simulated	100
Measure analyzed	radius of flight

Area of the mosquito actuation.



Area of the mosquito actuation.



Francisco Borges, Albert Gutierrez-Milla, Remo Suppi, Emilio Luque, Marylene de Brito Arduino. An Agent-Based Model for Assessment of Aedes Aegypti Pupal Productivity. WSC 2015. (CORE B)

Outline

Results: Communication patterns

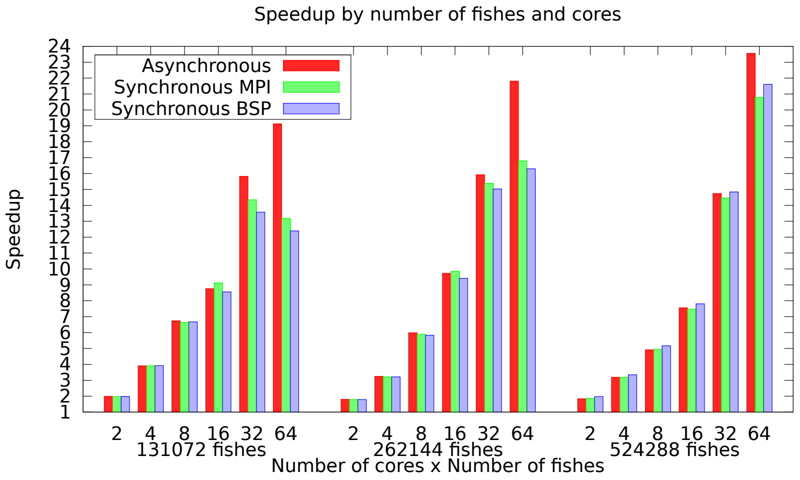
Objective

Speedups comparison of the communication strategies: asynchronous, synchronous BSP, and synchronous MPI.

Data and parameters

ABM	Fish schooling
Number of agents	131K, 262K and 524K
Number of cores	2, 4, 8, 16, 32 and 64
Measure analyzed	Speedup

Speedup of each communication strategy



Results: Communication patterns

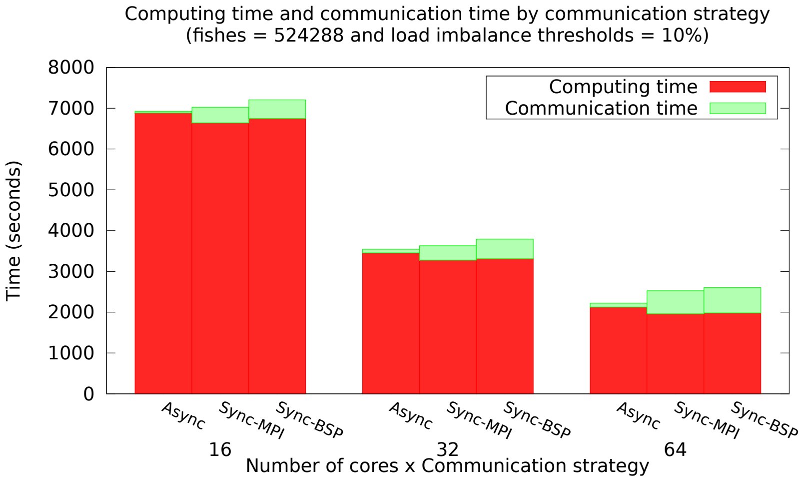
Objective

Computing time and communication time of communication strategies.

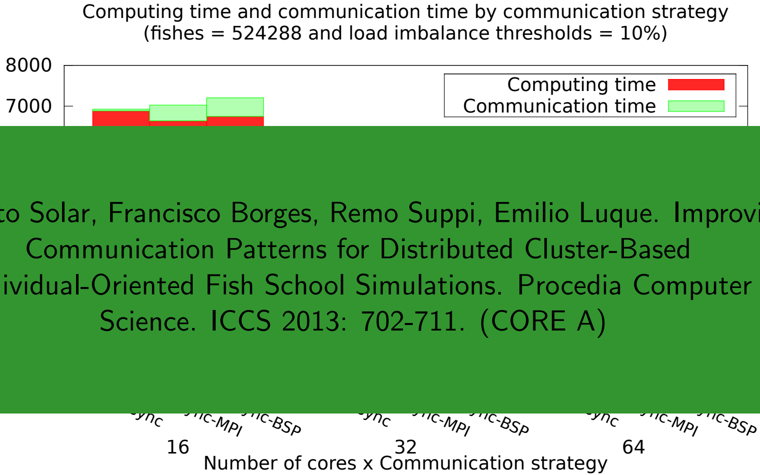
Data and parameters

ABM	Fish schooling
Number of agents	524K
Number of cores	16, 32 and 64
Measure analyzed	Computing and communication time (s)

Computing time and communication time x communication strategies



Computing time and communication time x communication strategies



Roberto Solar, Francisco Borges, Remo Suppi, Emilio Luque. Improving Communication Patterns for Distributed Cluster-Based Individual-Oriented Fish School Simulations. Procedia Computer Science. ICCS 2013: 702-711. (CORE A)

Results: Hybrid cluster-based partitioning

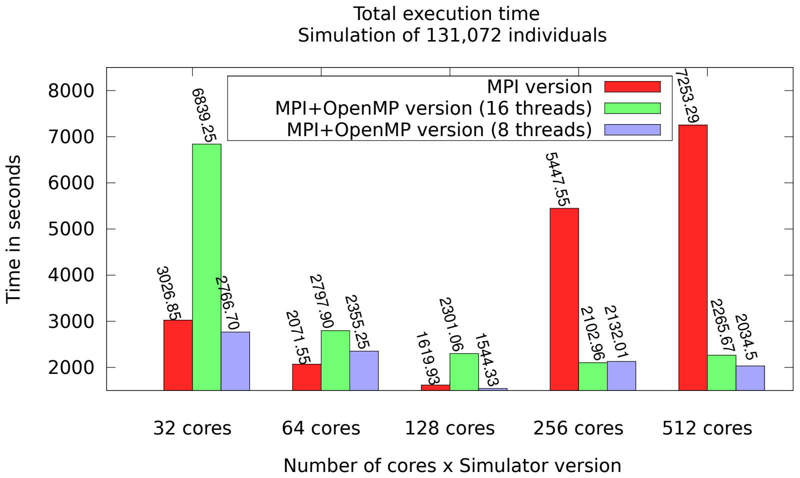
Objective

Total execution time comparison between the partitioning approaches: Pure MPI and Hybrid.

Data and parameters

ABM	Fish schooling
Number of agents	131K
Number of cores	32, 64, 128, 256 and 512
Number of threads	No threads, 8 and 16.
Measure analyzed	Execution time (s)

Total execution time comparison between the MPI and the MPI+OpenMP



Results: Hybrid cluster-based partitioning

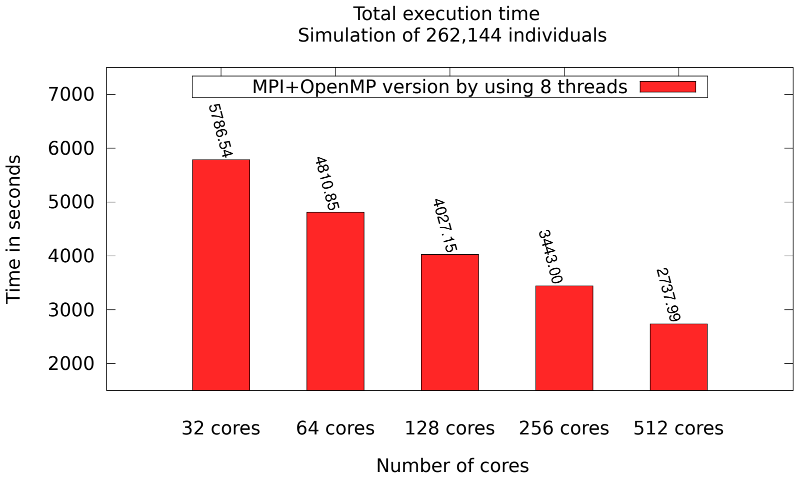
Objective

Scalability of the hybrid version.

Data and parameters

ABM	Fish schooling
Number of agents	262K
Number of cores	32, 64, 128, 256 and 512
Number of threads	8
Measure analyzed	Execution time (s)

Scalability of the hybrid version by using 8 threads per MPI process



Results: Hybrid cluster-based partitioning

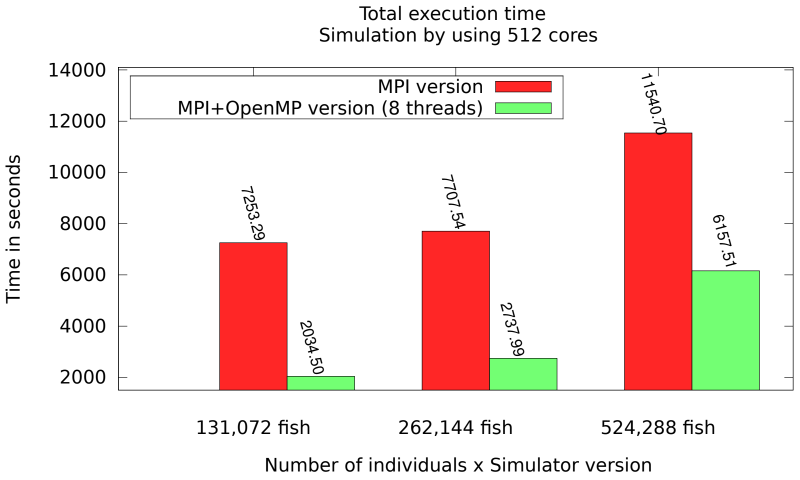
Objective

Total execution time comparison between the MPI and the MPI+OpenMP versions by using 512 cores.

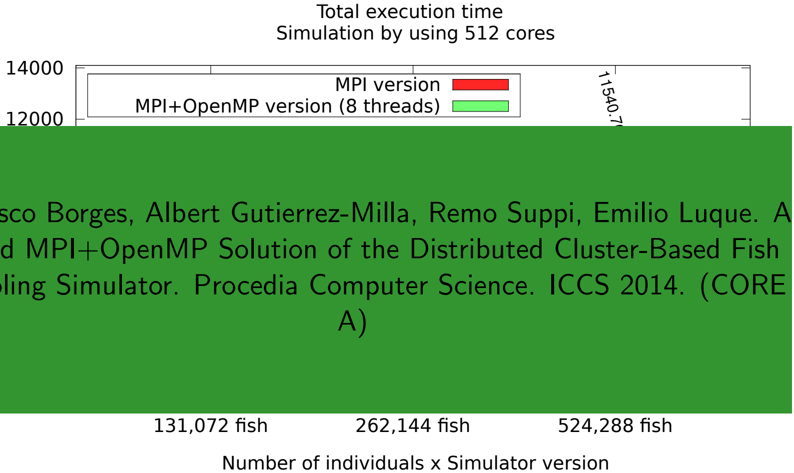
Data and parameters

ABM	Fish schooling
Number of agents	131k, 262K, 524k
Number of cores	512
Number of threads	8
Measure analyzed	Execution time (s)

Total execution time comparison between the MPI and the MPI+OpenMP versions by using 512 cores



Total execution time comparison between the MPI and the MPI+OpenMP versions by using 512 cores



Francisco Borges, Albert Gutierrez-Milla, Remo Suppi, Emilio Luque. A Hybrid MPI+OpenMP Solution of the Distributed Cluster-Based Fish Schooling Simulator. Procedia Computer Science. ICCS 2014. (CORE A)

Results: Strip partitioning

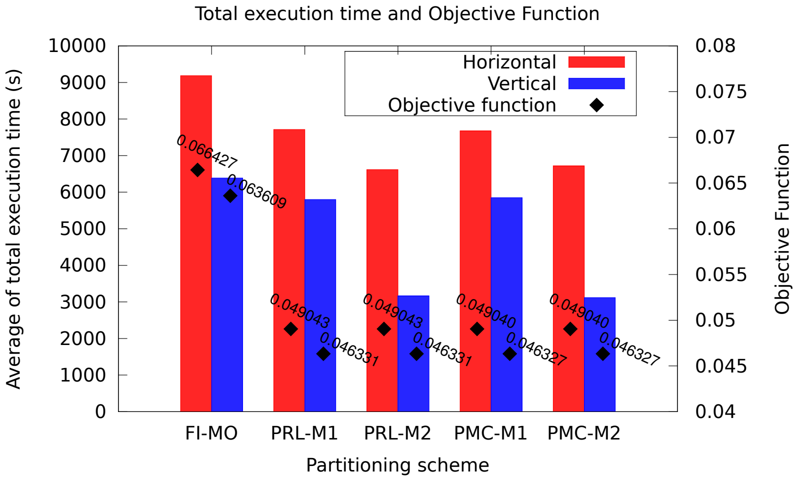
Objective

Total execution time average and objective function of each partitioning strategies.

Data and parameters

ABM	Ant colony
Number of agents	10k
Number of cores	64
Measure analyzed	Execution time (s)

Total execution time average and objective function of the partitioning strategies



Results: Strip partitioning

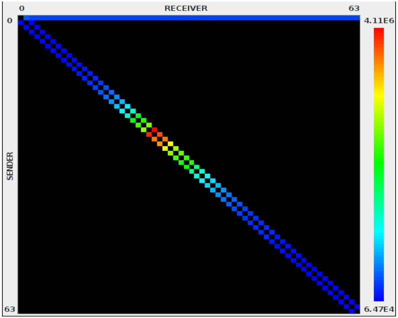
Objective

Comparison of total volume bytes of the worst and best strategies with heat maps.

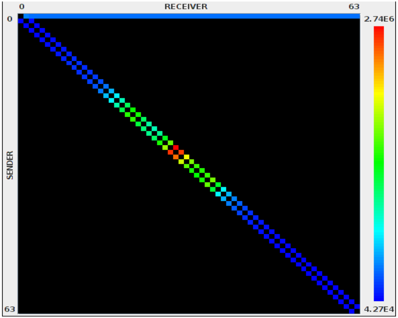
Data and parameters

ABM	Ant colony
Number of agents	10k
Number of cores	64
Strategies	best and worst
Measure analyzed	Total volume data

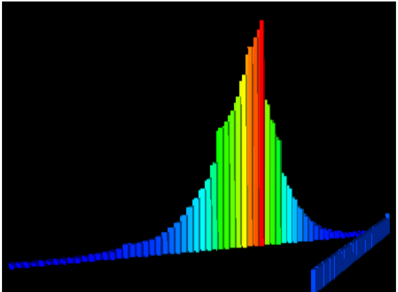
Comparison of total volume bytes with heat maps

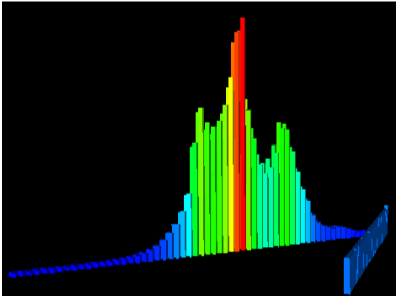


(a1) H-FI-M0



(b1) V-PMC-M2





Results: Strip partitioning

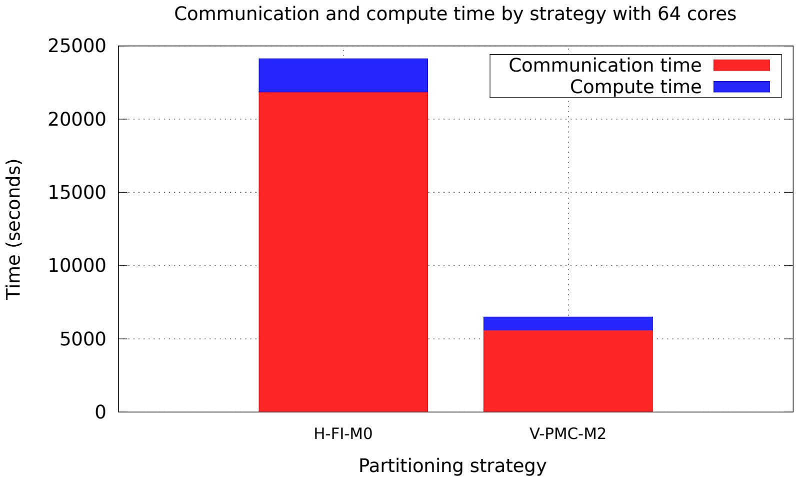
Objective

Communication and computing time of the worst and best strategies.

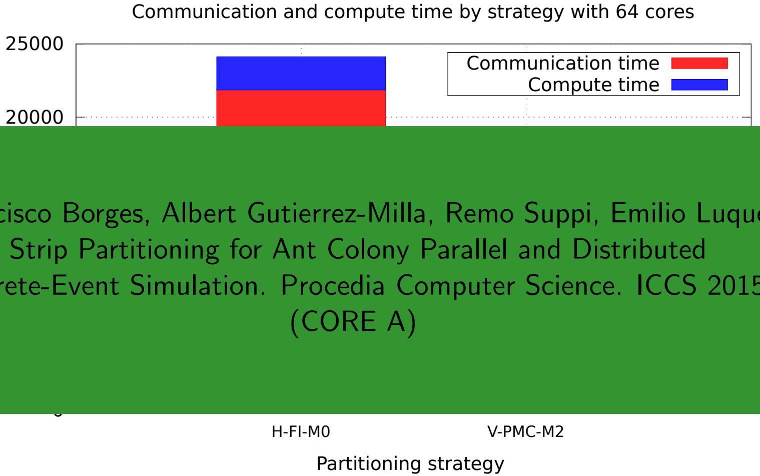
Data and parameters

ABM	Ant colony
Number of agents	10k
Number of cores	64
Strategies	best and worst
Measure analyzed	Execution time (s)

Communication and computing time of the worst and best strategies



Communication and computing time of the worst and best strategies



Francisco Borges, Albert Gutierrez-Milla, Remo Suppi, Emilio Luque. Strip Partitioning for Ant Colony Parallel and Distributed Discrete-Event Simulation. Procedia Computer Science. ICCS 2015. (CORE A)

Results: Hybrid Strip partitioning

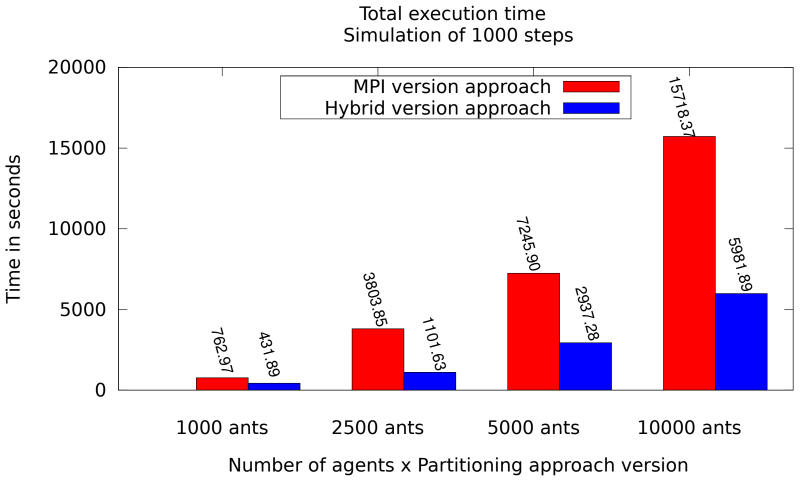
Objective

Total execution time of pure MPI and Hybrid strip partitioning algorithm.

Data and parameters

ABM	Ant colony
Number of agents	1k, 2.5k, 5k, 10k
Number of cores	64
Number of threads	No threads, [2..8]
Measure analyzed	Execution time (s)

Total execution time of pure MPI and Hybrid strip partitioning algorithm



Results: Hybrid Strip partitioning

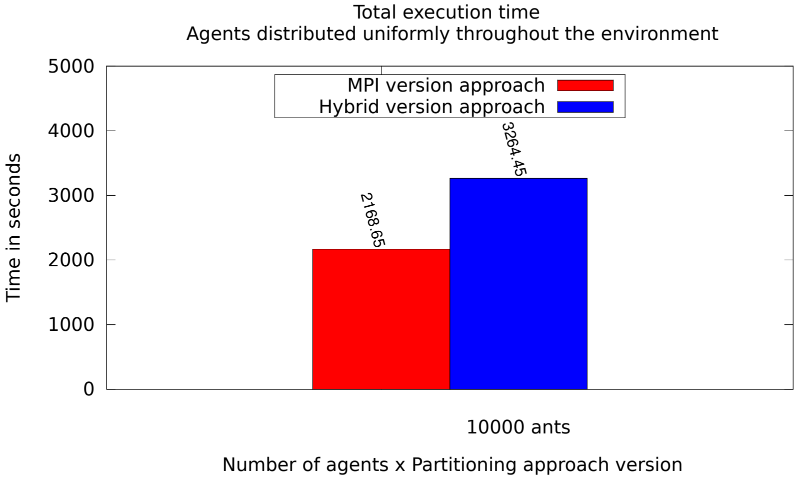
Objective

Total execution of pure MPI and hybrid strip partitioning algorithm. Agents distributed uniformly throughout the environment

Data and parameters

ABM	Ant colony
Number of agents	10k
Number of cores	64
Number of threads	No threads; [2..8] threads dynamically created.
Measure analyzed	Execution time (s)

Total execution of pure MPI and hybrid strip partitioning algorithm. Agents distributed uniformly throughout the environment.



Outline

Results: Agent layer

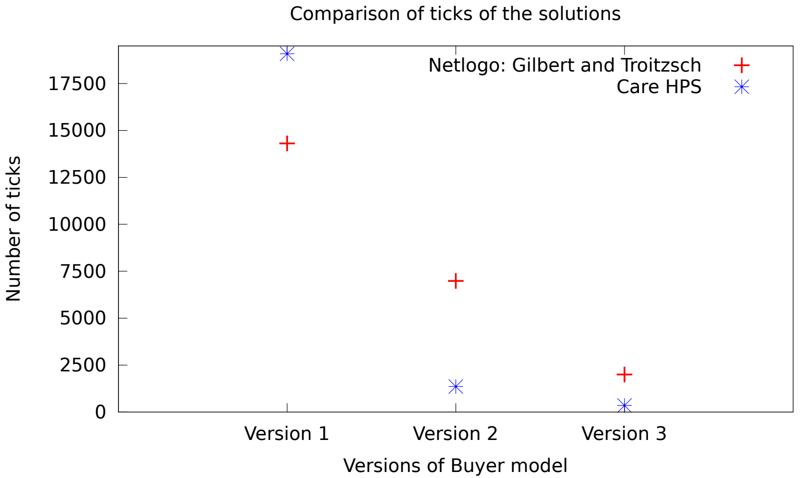
Objective

ABMS tools must be able to model agent rules and behaviors. So, this model can create collective and emergent behavior. It is important that these tools can reflect the interaction among agents

Data and parameters

ABM	Shopping agent
Number of agents	10
Number of cores	1
Size of list product	10
Number of stores	12
Measure analyzed	Number of ticks

Ticks of Gilbert and Troitzsch Netlogo version and Care HPS



Results: Environment layer

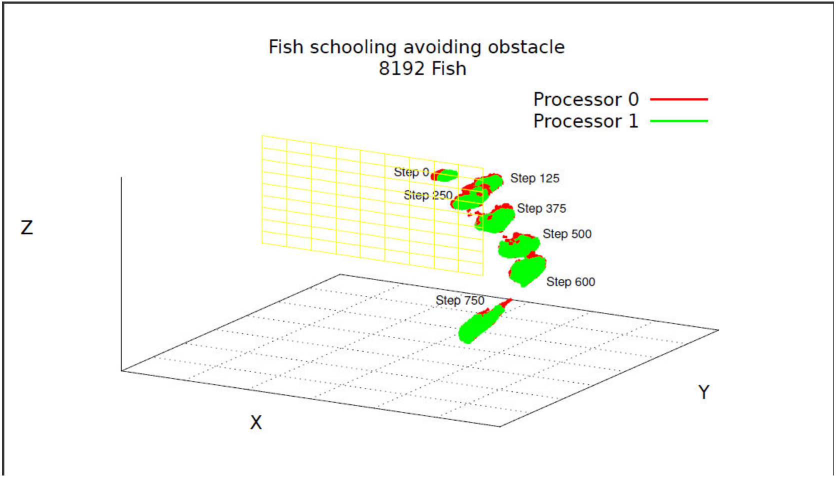
Objective

(1) Able to represent an environment and its objects using a math approach. (2) Enables the interaction between agents and the objects of the environment. (3) Enables the representation of simple rules in agents, and these simple rules can generate a collective behavior after agent interactions.

Data and parameters

ABM	Fish Schooling
Number of agents	8k
Number of cores	2
Analyzed	Interaction between agent and environment.

Fish repulsion behavior to avoid the collision. This experimentation was executed in two cores using 8192 agents



Results: Care HPS scalability

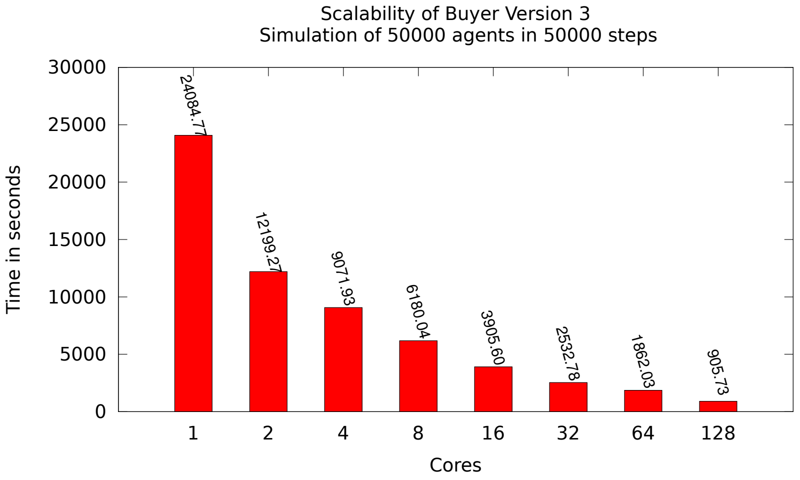
Objective

Present the scalability of Care HPS.

Data and parameters

ABM	Shopping agent
Number of agents	50K
Number of cores	1, 2, 4, 8, 16, 32, 64 and 128
Measure analyzed	Execution time (s)

Scalability of Buyer Version 3 with 50000 agents in 50000 steps



Results: Care HPS scalability

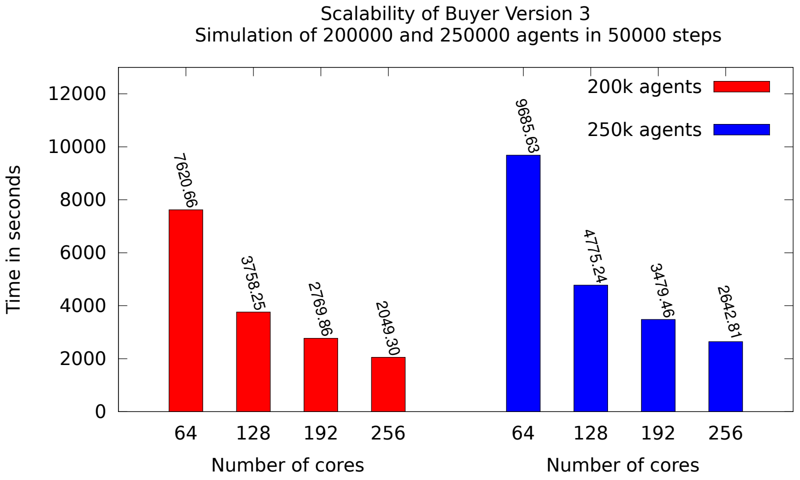
Objective

Present the scalability of Care HPS.

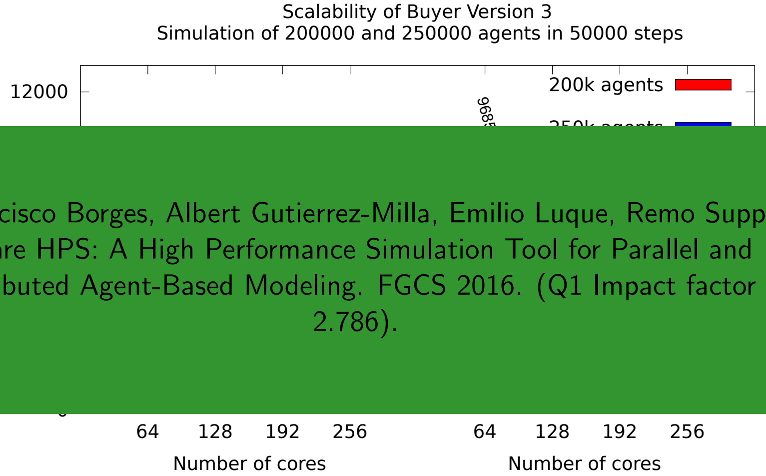
Data and parameters

ABM	Shopping agent
Number of agents	200K and 250K
Number of cores	64, 128, 192 and 256
Measure analyzed	Execution time (s)

Scalability of Buyer Version 3 in 50000 steps to 200k and 250k agents executed in 64, 128, 192 and 256 cores



Scalability of Buyer Version 3 in 50000 steps to 200k and 250k agents executed in 64, 128, 192 and 256 cores



Francisco Borges, Albert Gutierrez-Milla, Emilio Luque, Remo Suppi. Care HPS: A High Performance Simulation Tool for Parallel and Distributed Agent-Based Modeling. FGCS 2016. (Q1 Impact factor of 2.786).

Outline

The following papers are strictly related with this research.

- Roberto Solar, **Francisco Borges**, Remo Suppi, Emilio Luque. **Improving Communication Patterns for Distributed Cluster-Based Individual-Oriented Fish School Simulations**. ICCS 2013: 702-711. (CORE A)
- **Francisco Borges**, Roberto Solar, Remo Suppi y Emilio Luque. **Performance and scalability in distributed cluster-based individual-oriented fish school simulations**. Actas de las XXIV Jornadas de Paralelismo, Madrid (Madrid), 17-20 Septiembre 2013:401-406.
- Gallo, S., **Borges, F.**, Suppi, R., Luque Fadón, E., De Giusti, L. C., and Naiouf, M. (2013). **Mejoras en la eficiencia mediante Hardware Locality en la simulación distribuida de modelos orientados al individuo. In XVIII Congreso Argentino de Ciencias de la Computación**.
- **Francisco Borges**, Albert Gutierrez-Milla, Remo Suppi, Emilio Luque. **A Hybrid MPI+OpenMP Solution of the Distributed Cluster-Based Fish Schooling Simulator**. ICCS 2014. (CORE A)

- **Francisco Borges**, Albert Gutierrez-Milla, Remo Suppi, Emilio Luque. **Optimal Run Length for Discrete-Event Distributed Cluster-Based Simulations**. ICCS 2014. (CORE A)
- **Francisco Borges**, Albert Gutierrez-Milla, Remo Suppi, Emilio Luque. **Strip Partitioning for Ant Colony Parallel and Distributed Discrete-Event Simulation**. ICCS 2015. (CORE A)
- **Francisco Borges**, Albert Gutierrez-Milla, Remo Suppi, Emilio Luque, Marylene de Brito Arduino. **An Agent-Based Model for Assessment of Aedes Aegypti Pupal Productivity**. WSC 2015. (CORE B)
- **Francisco Borges**, Albert Gutierrez-Milla, Emilio Luque, Remo Suppi. **Care HPS: A High Performance Simulation Tool for Parallel and Distributed Agent-Based Modeling**. FGCS 2016. (Q1 Impact factor of 2.786).

The following papers are other publications with our research group related with ABM and HPC.

- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Individual-Oriented Model Crowd Evacuations Distributed Simulation**. ICCS 2014. (CORE A)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd evacuations SaaS: an ABM approach**. ICCS 2014. (CORE A)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Simulació de evacuaciones multitudinarias basadas en modelos orientados al individuo**. Actas de las XXV Jornadas de Paralelismo, Valladolid, 17-19 Septiembre 2014.
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd Dynamics Modeling and Collision Avoidance with OpenMP**. WSC 2015. (CORE B)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd turbulence with ABM and Verlet Integration on GPU cards**. ICCS 2016. (CORE A)

- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Individual-Oriented Model Crowd Evacuations Distributed Simulation**. ICCS 2014. (CORE A)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd evacuations SaaS: an ABM approach**. ICCS 2014. (CORE A)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Simulació de evacuaciones multitudinarias basadas en modelos orientados al individuo**. Actas de las XXV Jornadas de Paralelismo, Valladolid, 17-19 Septiembre 2014.
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd Dynamics Modeling and Collision Avoidance with OpenMP**. WSC 2015. (CORE B)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd turbulence with ABM and Verlet Integration on GPU cards**. ICCS 2016. (CORE A)

- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Individual-Oriented Model Crowd Evacuations Distributed Simulation**. ICCS 2014. (CORE A)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd evacuations SaaS: an ABM approach**. ICCS 2014. (CORE A)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Simulació de evacuaciones multitudinarias basadas en modelos orientados al individuo**. Actas de las XXV Jornadas de Paralelismo, Valladolid, 17-19 Septiembre 2014.
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd Dynamics Modeling and Collision Avoidance with OpenMP**. WSC 2015. (CORE B)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd turbulence with ABM and Verlet Integration on GPU cards**. ICCS 2016. (CORE A)

- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Individual-Oriented Model Crowd Evacuations Distributed Simulation**. ICCS 2014. (CORE A)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd evacuations SaaS: an ABM approach**. ICCS 2014. (CORE A)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Simulació de evacuaciones multitudinarias basadas en modelos orientados al individuo**. Actas de las XXV Jornadas de Paralelismo, Valladolid, 17-19 Septiembre 2014.
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd Dynamics Modeling and Collision Avoidance with OpenMP**. WSC 2015. (CORE B)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd turbulence with ABM and Verlet Integration on GPU cards**. ICCS 2016. (CORE A)

- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Individual-Oriented Model Crowd Evacuations Distributed Simulation**. ICCS 2014. (CORE A)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd evacuations SaaS: an ABM approach**. ICCS 2014. (CORE A)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Simulació de evacuaciones multitudinarias basadas en modelos orientados al individuo**. Actas de las XXV Jornadas de Paralelismo, Valladolid, 17-19 Septiembre 2014.
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd Dynamics Modeling and Collision Avoidance with OpenMP**. WSC 2015. (CORE B)
- Albert Gutierrez-Milla, **Francisco Borges**, Remo Suppi, Emilio Luque. **Crowd turbulence with ABM and Verlet Integration on GPU cards**. ICCS 2016. (CORE A)

Final considerations

7. Conclusion

8. Future work

Outline

- We introduce Care High Performance Simulation (HPS).
- The initial idea of Care HPS comes up as a methodology to support our research group, with the aim of answering the following question: how can we generalize our HPC techniques and approaches for agent-based models that demand high performance?
- Care HPS is a methodology to execute agent-based modeling and simulation in a parallel and distributed architecture.

- We introduce Care High Performance Simulation (HPS).
- The initial idea of Care HPS comes up as a methodology to support our research group, with the aim of answering the following question: how can we generalize our HPC techniques and approaches for agent-based models that demand high performance?
- Care HPS is a methodology to execute agent-based modeling and simulation in a parallel and distributed architecture.

- We introduce Care High Performance Simulation (HPS).
- The initial idea of Care HPS comes up as a methodology to support our research group, with the aim of answering the following question: how can we generalize our HPC techniques and approaches for agent-based models that demand high performance?
- Care HPS is a methodology to execute agent-based modeling and simulation in a parallel and distributed architecture.

- Care HPS is a scientific instrument to do research on HPC for agent-based models that demand high performance solutions. Care HPS enables both:
  - application area researchers to gain knowledge about the system under study using ABMs that require high performance computing solutions. This is possible because Care HPS offers well-defined and simple interfaces for this type of user in which all HPC complexity is hidden;
  - and, HPC expert users to develop approaches of high performance parallel and distributed simulation for ABM problems without high programming effort. Care HPS was projected using good object-oriented design practices which allow for the extension and reuse of the main HPS features.

- Care HPS is a scientific instrument to do research on HPC for agent-based models that demand high performance solutions. Care HPS enables both:
  - application area researchers to gain knowledge about the system under study using ABMs that require high performance computing solutions. This is possible because Care HPS offers well-defined and simple interfaces for this type of user in which all HPC complexity is hidden;
  - and, HPC expert users to develop approaches of high performance parallel and distributed simulation for ABM problems without high programming effort. Care HPS was projected using good object-oriented design practices which allow for the extension and reuse of the main HPS features.

- Care HPS is a scientific instrument to do research on HPC for agent-based models that demand high performance solutions. Care HPS enables both:
  - application area researchers to gain knowledge about the system under study using ABMs that require high performance computing solutions. This is possible because Care HPS offers well-defined and simple interfaces for this type of user in which all HPC complexity is hidden;
  - and, HPC expert users to develop approaches of high performance parallel and distributed simulation for ABM problems without high programming effort. Care HPS was projected using good object-oriented design practices which allow for the extension and reuse of the main HPS features.

- As part of our main findings and contributions, we present Care HPS, and we show through experimentation that Care HPS meets its objective and can be used as a scientific instrument for agent-based modeling that requires high performance parallel and distributed simulations.

Outline

7. Conclusion

8. Future work

- Currently, we are doing a comprehensive study of the ABM for the assessment of Aedes Aegypti pupal productivity.
- There are still room for improvement: components, models, HPC strategies, features.

- Currently, we are doing a comprehensive study of the ABM for the assessment of Aedes Aegypti pupal productivity.
- There are still room for improvement: components, models, HPC strategies, features.

Care HPS: A High Performance Simulation Methodology for Complex Agent-Based Models

Francisco Borges

Advisor: Dr. Remo Suppi

Federal Institute of Bahia
and
Universitat Autònoma de Barcelona
Computer Architecture & Operating Systems Department
February 21, 2017, Salvador

Care HPS: A High Performance Simulation Methodology for Complex Agent-Based Models

Francisco Borges

Advisor: Dr. Remo Suppi

Federal Institute of Bahia
and
Universitat Autònoma de Barcelona
Computer Architecture & Operating Systems Department
February 21, 2017, Salvador

franciscoborges@ifba.edu.br

Care HPS: A High Performance Simulation Methodology for Complex Agent-Based Models

Francisco Borges

Advisor: Dr. Remo Suppi

Federal Institute of Bahia
and
Universitat Autònoma de Barcelona
Computer Architecture & Operating Systems Department
February 21, 2017, Salvador

franciscoborges@ifba.edu.br

Thank you for your attention!