CFBM - A Framework for Data Driven Approach in Agent-Based Modeling and Simulation

Thai Minh Truong[1,2], Frédéric Amblard[1], Benoit Gaudou[1] and Christophe Sibertin Blanc[1]

[1]UMR 5505 CNRS-IRIT, Université Toulouse 1 Capitole, Toulouse, France
[2]CIT, Can Tho University, Can Tho, Vietnam
tmthai@cit.ctu.edu.vn, {frederic.amblard, benoit.gaudou, sibertin}@ut-capitole.fr

Abstract. Recently, there has been a shift from modeling driven approach to data driven approach in Agent Based Modeling and Simulation (ABMS). This trend towards the use of data-driven approaches in simulation aims at using more and more data available from the observation systems into simulation models [1, 2]. In a data driven approach, the empirical data collected from the target system are used not only for the design of the simulation models but also in initialization, evaluation of the output of the simulation platform. That raises the question how to manage empirical data, simulation data and compare those data in such agent-based simulation platform.

In this paper, we first introduce a logical framework for data driven approach in agent-based modeling and simulation. The introduced framework is based on the combination of Business Intelligence solution and a multi-agent based platform called **CFBM (Combination Framework of Business intelligence and Multi-agent based platform)**. Secondly, we demonstrate the application of CFBM for data driven approach via the development of a Brown Plant Hopper Surveillance Models (BSMs), where CFBM is used not only to manage and integrate the whole empirical data collected from the target system and the data produced by the simulation model, but also to initialize and validate the models.

The successful development of the CFBM consists not only in remedying the limitation of agent-based modeling and simulation with regard to data management but also in dealing with the development of complex simulation systems with large amount of input and output data supporting a data driven approach.

Keywords: Agent-Based Model, BI Solution, Brown Plant Hopper, Data Driven Approach, Data Warehouse, Multi-Agent Based Simulation.

1. Introduction

Today, the agent-based simulation approach is increasingly used to develop simulation systems in quite different fields such as: (1) in natural resources management – e.g., an agent-based simulation system consisting of a set of management processes (i.e. water, land, money, and labour forces) built to simulate catchment water management in the north of Thailand [3] or the agent-based modeling used to develop a water resource management and assessment system which combines spatiotemporal models of ecological indicators such as rainfall and temperature, water flow and plant growth [4]; (2) in biology - a model for the study of epidemiology or evacuation of injured persons [5–8]; and (3) in sociology - a multi-agent system to discover how social actors could behave within an organization [9] or an agent-based simulation model to explore rules for rural credit management [10].

In the building such systems, we are not only concerned with modeling driven approach – that is how to model and combine coupled models from different scientific fields - but also with data driven approach – that is how to use empirical data collected from the target system in modeling, simulation and analysis [1, 11–13]. Such systems mainly take the form of empirical data gathered from the target system and these data can be used in processes such as design, initialization, calibration and validation of models. That raises the question about which framework to deal with data driven approach that can help to manage empirical data, simulated data and compare those data in agent-based simulation systems as mentioned above.

In this paper, the first introduction is the general architecture for data driven approach which is called CFBM (the Combination Framework of Business Intelligence solution and Multi-agent platform). The CFBM in order to adapt "the logic of simulation" proposed in [13, 14] and it could serve the following purposes: model and execute multi-agents simulations, manage the input and output data of simulations, integrate data from different sources and enable to analyze high volume of data. We argue that BI (Business Intelligence) solutions are a good way to handle and analyze big datasets. Because a BI solution contains a data warehouse, integrated data tools (ETL, Extract-Transform-Load tools) and Online Analytical Processing tools (OLAP tools), it is well adapted to manage, integrate, analyze and present huge amounts of data [15, 16]. The second is an demonstration the application of CFBM into develop Brown plant hopper Surveillance network Models (BSMs) by using data driven approach.

2. Related works

Technologies or tools from different fields used within integrated systems become a common trend today. In particular, for the development of decision support systems or prediction systems based on simulation approaches, data warehouse (DW), online analytical processing (OLAP) technologies and simulation often represent a good solution. Data warehouse and analysis tools as a BI solution can help users to manage a large amount of simulation data and make several data analyses that support the decision-making processes [17] and [18]. The combination of simulation tools and DW is increasingly used and applied in different areas. In [19], Madeira et al. proposed a new approach dedicated to the analysis and comparison of large amounts of output data from different experiments or from similar experiments across different systems. Data warehouse and OLAP tools were recommended for collecting and analyzing simulation results of a system [20]. Although [19][20] are only applications of OLAP technologies to a special problem, these works demonstrate that a multidimensional database is suitable to store several hundreds of thousands of simulation results. Simulation models, DW and analysis tools with OLAP technologies were also involved in decision support systems or forecast systems [21][22]. Not only their researches solve specific problems but they also demonstrated a promising use of gathering and analyzing simulation results by using data warehouse and OLAP technologies. In [23], Mahboubi et al. also use data warehouse and OLAP technologies to store and analyze a huge amount of output data generated by the coupling of complex simulation models such as biological, meteorological and so on. In particular, they propose a multidimensional data schema of a data warehouse for storing and analyzing simulation results.

The state of the art demonstrates therefore the practical possibility and the usefulness of the combination of simulation, data warehouse and OLAP technologies. It also shows the potential of a general framework that has, as far as we are aware, not yet proposed in the literature.

3. A Framework for Data Driven Approach in Agent-Based Simulation

In this Section, I present a solution addressing the requirements concerning data (collected data from target system and simulated data) management, initialization (used collected data as input of simulation model) and analysis (e.g. validation the output of simulation) in data driven approach for agent-based simulation[24]. First, we introduce general architecture could serve the following purposes: to model and execute multi-agents simulations, manage input and output data of simulations, integrate data from different sources and enable to analyze high volume of data. The second, I introduce an implementation of the logical framework in the GAMA multi-agent based simulation platform

3.1. A Logical Framework to Manage and Analyze Data for Agent-based Models

The Combination Framework of Business Intelligence solution and Multi-agent platform (CFBM) is designed based on our definition of an extended computer simulation system[25]. We have therefore designed CFBM with four major components: (1) model design tool; (2) model execution tool; (3) execution analysis tool; and (4) database management tool. CFBM is a logical framework to manage and analyze data in agent-based modeling and its architecture is summarized in Figure 3.

Simulation system

The simulation system includes at the same time the model design (that we won't detail at this stage) and the model execution parts. It executes simulations and handles their input/output data. From a data management point of view, it plays the role of an OnLine Transaction Processing (OLTP) or of an operational source system. It is considered as an outside part of the data warehouse [26].

Three layers with five components compose the simulation system. The **simulation interface** is a user environment that helps the modeler to design and implement his models, execute simulations and visualize some observations selected (through the design of the graphical interface) from the simulation. **Multi-agent simulation models** are used to simulate phenomena that the modeler aims at studying. The **SQL-agent** plays the role of the database management tool and can access a relational database**.** They can be considered as intermediary in between the simulation system and the data management part. It is a particular kind of agents that supports Structured Query Language (SQL) functions to retrieve simulation inputs from the database containing empirical data (Empirical database) as well as parameter values and scenarios (Simulation database). SQL-agent is able to store output simulation data. SQL-agent are also able to transform data (in particular the data type) from simulation model to relational database, and conversely. Empirical database and Simulation database are relational databases. **Empirical database** is used to store empirical data gathered from the target system that are needed for the simulation and analysis phases. **Simulation database** is used to manage the simulation models, simulation scenarios and output results of the simulation models. These two data sources (Empirical as well as Simulation databases) will be used to feed the second part of the framework, namely the Data warehouse system.

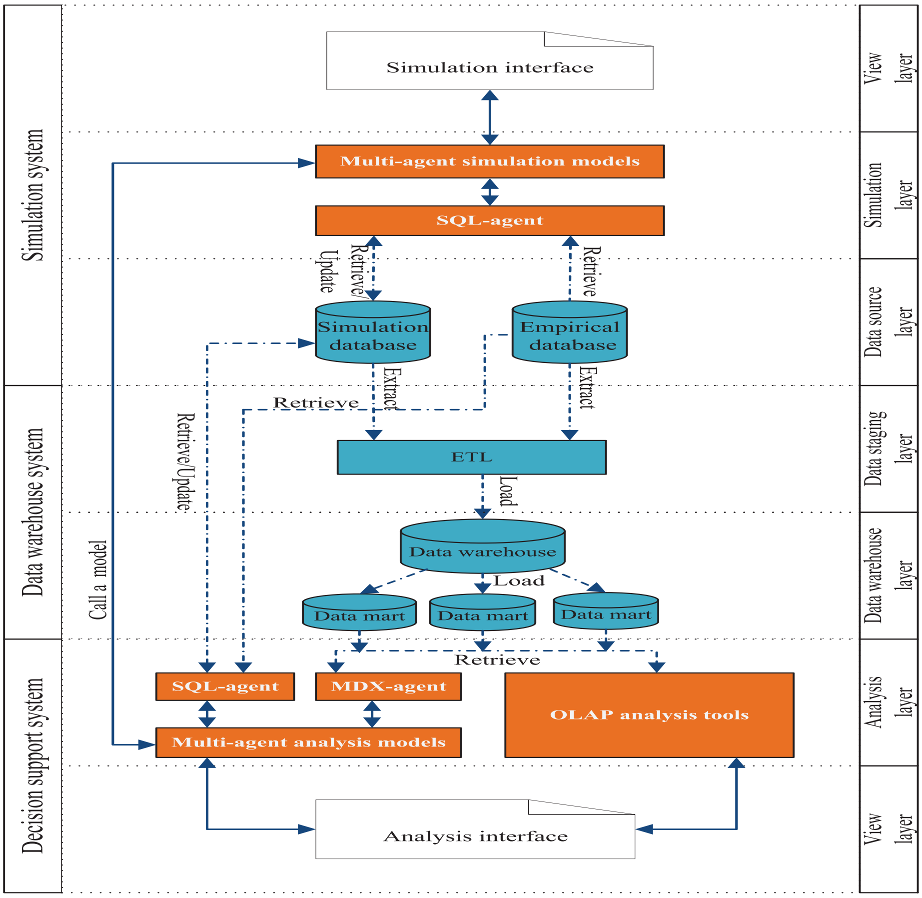

Figure 3: CFBM architecture

Note:

DW	: Data Warehouse
SQL	: Structured Query Language
OLAP	: OnLine Analysis Processing

ETL : Extract – Transform – Load
MDX : Multi Dimensional eXpressions
SQL-agent : Agent supporting SQL features to query data.
MDX-agent: Agent supporting MDX features to query data.
$\dashrightarrow$: Data flow.
$\longrightarrow$: Intercommunication

Data warehouse system

The data warehouse system is crucial in our approach as it enables to integrate data from different sources (simulation data as well as empirical data from the target systems). It is also used as data store to provide data to the decision support systems. The data warehouse system is divided into three parts. The **ETL** (Extract-Transform-Load) is a set of processes with three responsibilities: it extracts all kind of data (empirical data and simulation data) from the simulation system; then, ETL transforms the extracted data into an appropriate data format; finally, it loads the transferred data into the data warehouse. The **data warehouse** is a relational database used to store historical data loaded from simulation systems and from other sources (empirical sources for instance) by the ETL. The **data mart** is a subset of data stored in the data warehouse used as a data source for the concrete analysis requirement. We can create several data marts depending on the analysis requirements. The data mart is a multidimensional database, which is designed based on multidimensional approach. Therefore, the structure of the data mart is presented using star join, fact tables and dimension tables to present. Thanks to its multidimensional structure, data marts are particularly useful to help users to improve the performance of analytic processes.

Decision support system

In CFBM, the decision support system component is a software environment supporting analysis, visualization of results and decision-making. In our design, we propose to use either existing OLAP analysis tools or a multi-agent platform equipped with analysis features or a combination of both. The decision support system of CFBM is composed of four parts. The **analysis interface** is a graphical user interface used to handle analysis models and visualize results. The **multi-agent analysis models** are a set of tools dedicated to the analysis of multi-agent simulations, they are created based on analysis requirements and handled via the analysis interface. The **MDX-agent** is a special kind of agents, which supports MultiDimensional eXpressions (MDX) functions to query data from a multidimensional database. The MDX-agent serves as a bridge in between multi-agent analysis models and data marts. It is used to retrieve data from the data marts into the data warehouse system. The **OLAP analysis tools** could be any analysis software that supports OLAP operators. The multi-agent analysis tools may need several kind of retrievals such as: (1) to retrieve detailed data from relational databases (simulation database or empirical database) for analyses for instance for comparison in-between models or in-between a model output and gathered empirical data; (2) to retrieve pre-aggregated data from multidimensional databases (data marts) for multi-scale analysis. Hence in the decision support system, we designed two kinds of database access agent: on the one hand SQL-agent uses structured query language to retrieve data from relational database and on the other hand MDX-agent uses multidimensional expressions to query data from multidimensional database. Therefore, the multi-agent analysis tools can also use SQL-agents (same SQL-agents as in the simulation system) or MDX agents to appropriately access data.

3.2. Implementation of CFBM with the GAMA Platform

The CFBM has been implemented in GAMA[1] within a three-tier architecture [25]: (1) the **Presentation tier** plays the role of the view layer of simulation system and decision support system in the CFBM architecture; (2) the **Logic tier** coordinates the application commands, as it plays the role of both simulation and analysis layers in the CFBM architecture; and (3) the **Data tier** plays two roles in the framework: data source layer and data warehouse layer. The main functions of this tier are to store and retrieve data from a database or a file system using JDBC and OLAP4J[2].

Thanks to added database features of GAMA, we can create agents and define the environment of the simulation by using data selected from database, store simulation results into relational databases or query data from multidimensional database. Database features are implemented in the *irit.gaml.extensions.database* GAMA plug-in with the following features:

- Agents can execute SQL queries (create, insert, select, update, drop, delete) to various kinds of DBMS.
- Agents can execute MDX (Multidimensional Expressions) queries to select data from multidimensional objects such as cubes, and return multidimensional cell sets that contain the cube's data[3].

These features are implemented in two kinds of components: *skills* (SQLSKILL, MDXSKILL) and agent (AgentDB). They help us to gain flexibility in the management of simulation models and the analysis of simulation results. In this part, we do not demonstrate functions, which have been implemented in GAMA. More details are presented in website of GAMA[4].

4. Applying CFBM to develop Brown Plant Hopper Surveillance Models

This Section demonstrates an application of CFBM to: (1) manage input and output data of pest surveillance models called Brown plant hopper Surveillance Models (BSMs). The BSMs was built to predict the invasion of Brown Plant Hoppers (BPHs) on the rice fields based on the data collected from the network light traps network in

[1] https://github.com/gama-platform/gama/wiki
[2] http://www.olap4j.org/
[3] http://msdn.microsoft.com/en-us/library/ms145514.aspx
[4] https://github.com/gama-platform/gama/wiki/G__UsingDatabase

the Mekong Delta River in Vietnam. The models are part of the research project JEAI-DREAM project[5] and their methodology and implementations were presented in [27]; (2) initialization, using gathered data from target system as input data of BSMs; and validation the output of BSMs.

4.1. Data management for the models

In BSMs, an integrated model of an environment surveillance network on an insect ecosystem is built for an application at the regional scale. The input data (empirical data) of this model are divided into four main groups: Insect sampling data (daily sampling data of about 10 insect species at more than 300 light-traps in the Mekong Delta region of Vietnam), Administrative regions (province, district and small town scales), Land uses (seasonal crops of multiple tree species of the region), Natural phenomena (wind, temperature, humidity, etc).

In experiments, we tested the BSMs with various scenarios and several replications for each scenarios. We also classified the outputs of simulations into two groups: (1) number of insects in light traps and (2) mean density of insects for small towns. The Figure 4 illustrates a part of the entity relationship diagram of input/out data of the BSMs.

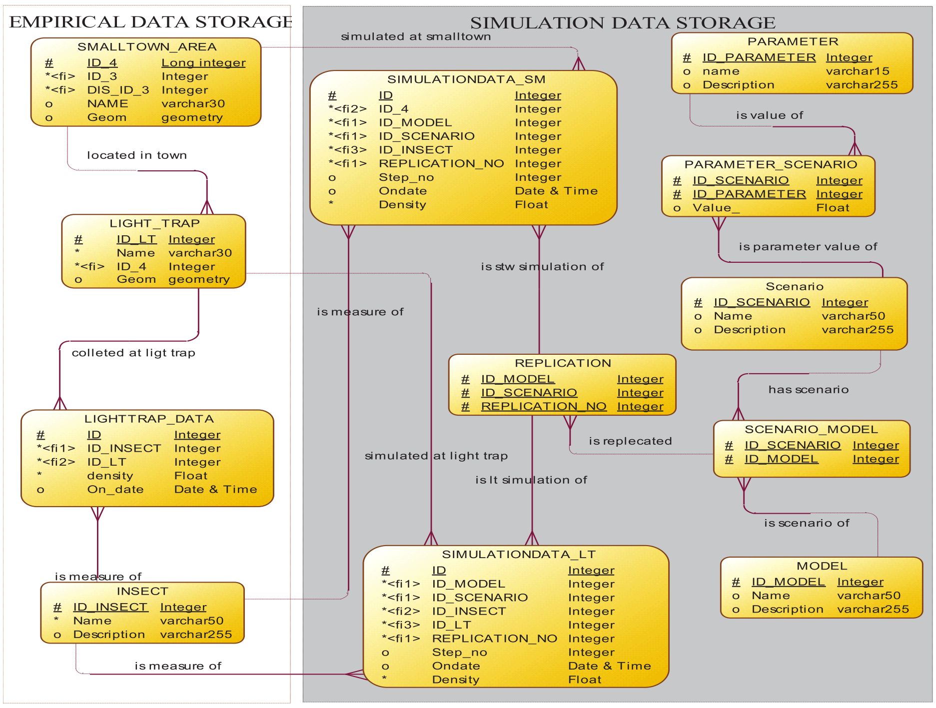

Figure 4: Entity relationship diagram of BSMs

---

[5] http://www.vietnam.ird.fr/les-activites/renforcement-des-capacites/jeunes-equipes-aird/jeai-dream-umi-209-ummisco-2011-2013

In the case of a development of the model without using CFBM, for an example with only one insect species in three provinces as in our scenarios in [25], we need to manually filter the data by some tools and the selected data can be stored in some individual files. These files will be used as the input of the model. With CFBM, these works are just replaced by some queries applied on database. This flexibility allows the modelers to easily change their temporal or spatial scale of their scenarios. In other words, CFBM supports to select and update the appropriate data at every micro scale of the model for every simulation step. Furthermore, the input and output data are also distributed and portable

4.2. Initialization and validation of BSMs

By applying CFBM, all kind of data of the BSMs are managed by a relational database management system, which makes it more easy to integrate and choose data for each scenario, calibration and validation without manipulating external tools [28]. For example, we can select data and define environment in Example 1 and agents for simulating the invasion of BPHs on the rice fields of three provinces (Dong Thap, Soc Trang and Bac Lieu) as the codes in Example 2; and if we want to add more provinces, then we only change the "WHERE" condition. Hence CFBM in GAMA not only helps modelers in the management of data but it also helps to build agent-based models in a more flexible way by parameterizing the inputs of the models.

Example 1: Specification of the boundary of simulation

```
String PROVINCES<- ' (38254, 38257,38249)';//Dong Thap, Soc Trang, Bac Lieu
map<string,string> BOUNDS <-
        ['host'::'localhost','dbtype'::'postgres','database'::'SurveillanceNetDB'
        ,'port'::'5432','user'::'postgres','passwd'::'acb'
      ,'select'::'SELECT ST_AsBinary(geom) as geo from PROVINCE_AREA'
      + 'WHERE ID_2 in '+ PROVINCES];
```

Example 2: Select appropriate data from database and create agents

```
//      Specify light trap
String NODE<-
        "SELECT
                id_lt , lt.name as lighttrap,
                regions_area.id_1 as id_1, province_are.id_2 as id_2,
                distric_area.id_3 as id_3, smalltown_area.id_4 as id_4,
                province_are.name as province, distric_area.name as district,
                smalltown_are.name as smalltown,  st_asbinary(lt.geom) as geo "
        FROM
                regions_area, province_are, distric_area, smalltown_area,
                light_trap as lt
        WHERE
                 (province_are.id_1=regions_area.id_1) and
```

```
                    (district_are.id_2=province_area.id_2) and
                    (smalltown_are.id_3=district_area.id_3) and
                    (lt.id_4 =smalltown.id_4) and
                    province_area.id_2 in '"+ provinces;
...
//Create species
create species: db number: 1
{
      ...
  create species: node
      from: list(self select [params:: PARAMS, select:: NODE])
      with:[ id :: 'ID', name :: 'LightTrap', district_name :: 'District',
                    province_name :: 'Province', id_0 :: 'ID_0', id_1 :: 'ID_1',
                    id_2 :: 'ID_2',shape::'geo'];
      ...
}
```

Thanks to the combination of BI solution and multi-agent based simulation platform, CFBM can help us to integrate empirical and simulated data (cf. Section 3). Because the simulation results and empirical data are already integrated in the data warehouse, we can define the data marts in accordance to the change of aggregation requirements without reproducing the results of the simulation models. Furthermore, agent model analysis may become easier and modelers can conduct several analyses on the integrated data such as comparing simulated data between models, calibrating and validating models, or aggregating high volumes of data by using database features in GAMA[28]. For example, we can validate the output of BSMs by calculating the similarity coefficients such as the RMSE between the collected data and simulation data (Example 3).

Example 3: calculate RMSE by model, scenario and replication_no

```
string query_str <-
"SELECT  ml.model, ml.scenario, ml.replication_no,
sqrt(avg(square(lt.simulation_value - lt.collected_value)))as RMSE
FROM lighttrapdata_facts as lt, models_dim as ml
WHERE  lt.id_models=ml.id_models
GROUPBY ml.model, ml.scenario, ml.replication_no";
ask db
{
   list<list> rmse_list <- list<list> (self select(params: PARAMS,
               select: query_str));
}
```

There are some flexibility of data integration and aggregation via CFBM such as reducing the complex structure in programming and aggregating high volumes of data. By applying CFBM, modelers can split the complex simulation models aiming at simulating a phenomenon and analyzing the simulation outputs into two separate models (SIMULATION MODEL and ANALYSIS MODEL), one for simulating the phenomenon and another for analyzing the outputs of simulation [28].

5. Conclusion

The most important of CFBM is a logical framework dealing with data-adapted computer simulations, including four major tools: (1) a *model design tool*: a software environment that supports a modeling language and a user interface, and this is generic enough to model any kind of system; (2) a *model execution tool*: a software environment that can run models; (3) an *execution analysis tool*: a software environment that supports statistical analysis features for the analysis of the simulation output; and (4) a *database tool*: a software environment that supports appropriate database management features for all components in the system.

In addition, CFBM is the powerful integration of a data warehouse, OLAP analysis tools and a multi-agent based platform. CFBM is useful to develop complex simulation systems with large amount of input/output data such as a what-if simulation system, a prediction/forecast system or a decision support system.

In practice, we recognize that CFBM allows the handling of complex simulation system. In particular, we can build several models to simulate the same phenomenon, conduct a lot of simulations for each of them and compare these simulation results (e.g. to determine which one is better for which parameter value domain). In this case, it is very difficult for modelers to manage, analyze or compare the output data of simulations if they do not have an appropriate tool. With the embedding of SQL agents and MDX agents in the simulation system of CFBM, modelers can create a database to manage and store simulation models, scenarios and outputs of simulation models more easily. In our case study, we have fully handled the input/output simulation data of the models. Selecting and comparing the output data between different replications or between the simulation data and the empirical data can be done in an easier and more efficient way.

References

1. Edmonds, B., Moss, S.: From KISS to KIDS – an " anti-simplistic " modelling approach. Multi-Agent Multi-Agent-Based Simul. 3415, 130–144 (2005).

2. Hassan, S.: Towards a Data-driven Approach for Agent-Based Modelling : Simulating Spanish, (2009).

3. Becu, N., Perez, P., Walker, A., Barreteau, O., Page, C.L.: Agent based simulation of a small catchment water management in northern Thailand. Ecol. Modell. 170, 319–331 (2003).

4. Gaudou, B., Sibertin-Blanc, C., Therond, O., Amblard, F., Arcangeli, J.-P., Balestrat, M., Charron-Moirez, M.-H., Gondet, E., Hong, Y., Louail, T., Mayor, E., Panzoli, D., Sauvage, S., Sanchez-Perez, J.-M., Taillandier, P., Nguyen, V.B., Vavasseur, M., Mazzega, P.: The maelia multi-agent platform for integrated assessment of low-water management issues. In: MABS, Multi-Agent-Based Simulation XIV-International Workshop (2013).

5. Rao, D.M., Chernyakhovsky, A., Rao, V.: Modeling and analysis of global epidemiology of avian influenza. Environ. Model. Softw. 24, 124–134 (2009).

6. Stroud, P., Valle, S. Del, Sydoriak, S., Riese, J., Mniszewski, S.: Spatial dynamics of pandemic influenza in a massive artificial society. Artif. Soc. Soc. Simul. 10, (2007).

7. Amouroux, E., Desvaux, S., Drogoul, A.: Towards virtual epidemiology: an agent-based approach to the modeling of H5N1 propagation and persistence in North-Vietnam. Intell. agents multi-agent Syst. 5357, 26–33 (2008).

8. Dunham, J.: An agent-based spatially explicit epidemiological model in MASON. J. Artif. Soc. Soc. Simul. 9, (2005).

9. Sibertin-Blanc, C., Roggero, P., Adreit, F., Baldet, B., Chapron, P., El-Gemayel, J., Mailliard, M., Sandri, S.: SocLab: A Framework for the Modeling , Simulation and Analysis of Power in Social Organizations. J. Artif. Soc. Soc. Simul. 16, (2013).

10. Barnaud, C., Bousquet, F., Trebuil, G.: Multi-agent simulations to explore rules for rural credit in a highland farming community of Northern Thailand. Ecol. Econ. 66, 615–627 (2008).

11. Hassan, S., Antunes, L., Pavon, J.: Mentat: a data-driven agent-based simulation of social values evolution. In: Multi-Agent-Based Simulation X. pp. 135–146. Springer Berlin Heidelberg (2010).

12. Hassan, S., Pavon, J., Gilbert, N.: Injecting Data into Simulation : Can Agent-Based Modelling Learn from Microsimulation ? In: The World Congress of Social Simulation 2008. , Washington, D.C (2008).

13. Hassan, S., Pavón, J., Antunes, L., Gilbert, N.: Injecting Data into Agent-Based Simulation. In: Simulating Interacting Agents and Social Phenomena. pp. 177–191. Springer Japan (2010).

14. Gilbert, N., Troitzsch, K.G.: Simulation for the Social Scientist. Open University Press (2005).

15. Mahboubi, H., Faure, T., Bimonte, S., Deffuant, G., Chanet, J.-P., Pinet, F.: A Multidimensional Model for Data Warehouses of Simulation Results. Int. J. Agric. Environ. Inf. Syst. 1, 1–19 (2010).

16. Vasilakis, C., El-Darzi, E.: A data warehouse environment for storing and analyzing simulation output data . Simul. Conf. (2004).

17. Inmon, W.H.: Building the Data Warehouse. Wiley Publishing Inc (2005).

18. Kimball, R., Ross, M.: The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling. John Wiley & Sons, Inc (2002).

19. Madeira, H., Costa, J.P., Vieira, M.: The OLAP and data warehousing approaches for analysis and sharing of results from dependability evaluation experiments. In: International Conference on Dependable Systems and Networks. pp. 86–99 (2003).

20. Sosnowski, J., Zygulski, P., Gawkowski, P.: Developing Data Warehouse for Simulation Experiments. In: Rough Sets and Intelligent Systems Paradigms. pp. 543–552. Springer Berlin Heidelberg (2007).

21. Ehmke, J.F., Grosshans, D., Mattfeld, D.C., Smith, L.D.: Interactive analysis of discrete-event logistics systems with support of a data warehouse. Comput. Ind. 62, 578–586 (2011).

22. Vasilakis, C., El-Darzi, E., Chountas, P.: A decision support system for measuring and modelling the multi-phase nature of patient flow in hospitals. In: Intelligent Techniques and Tools for Novel System Architectures. pp. 201–217. Springer Berlin Heidelberg (2008).

23. Mahboubi, H., Faure, T., Bimonte, S., Deffuant, G., Chanet, J.P., Pinet, F.: A Multidimensional Model for Data Warehouses of Simulation Results. Int. J. Agric. Environ. Inf. Syst. 1, 1–19 (2010).

24. Hassan, S., Antunes, L., Pavon, J., Gilbert, N.: Stepping on Earth: A Roadmap for Data-driven Agent-Based Modelling. In: the 5th Conference of the European Social Simulation Association (ESSA08). pp. 1–12 (2008).

25. Truong, T.M., Truong, V.X., Amblard, F., Sibertin-blanc, C., Drogoul, A., Le, M.N.: An Implementation of Framework of Business Intelligence for Agent-based Simulation. In: The 4th International Symposium on Information and Communication Technology (SoICT 2013). pp. 35–44. ACM (2013).

26. Kimball, R., Ross, M.: Data warehouse Toolkit: The complete guide to Dimentional Modeling. John Wiley & Sons, Inc (2002).

27. Truong, V.X.: Optimization by simulation of an environmental surveillance network - application to the fight against rice pests in the mekong delta (vietnam), (2014).

28. Truong, T.M., Amblard, F., Benoit, G., Sibertin-blanc, C.: To Calibrate & Validate an Agent-Based Simulation Model An Application of the Combination Framework of BI solution & Multi-agent platform. In: TThe 6th International Conference on Agents and Artificial Intelligence (ICAART 2014). pp. 172–183 (2014).