Models, Abstractions and Phases in Multi-Agent Based Simulation

Mizar Luca Federici, Stefano Redaelli and Giuseppe Vizzari
Dept. of Computer Science, Systems, and Communication
University of Milan–Bicocca
Milan, Italy
{mizar,redaelli,vizzari}@disco.unimib.it

***Abstract*— This paper introduces some considerations about simulation practice and a schema of the models that are implicitly and explicitly involved in a Multi-Agent Based Simulation (MABS). The aim of this work is to set simulation inside scientific framework. In order to do that we give an interpretation of the levels that compound a simulation and that constitute different kinds of abstraction. A clear awareness of the relations that exist between these levels and the corresponding steps, in fact, it is necessary if MABS wants to be adopted as a scientific investigation method. Our opinion is that this analysis suggests some answers to the objections that are often directed towards the use of simulation in scientific practice but also underlines some criticalities in this process.**

I. Introduction

Why do we use computer simulations? Winsberg in [1] claims that "simulations are often performed to investigate systems for which data are sparse". An assumption that stands behind the use of simulation as a tool to investigate reality is that reality cannot be known analyzing what can be considered as its single parts or components, but only by recreating it from its components. Assertors of the efficacy of the scientific use of simulation claim that a simulation expresses a theory, and that a theory expressed by a simulation produces a series of empiric predictions that can be directly compared to reality. This consideration lays on the conviction that a simulation is also a way to build up a scientific theory as it allows to check immediately the consequences of our assumptions and gives us precise indications for the correction or reformulation of a theory. Computer extends our calculus and memory capacity, and simulations that run on a computer are considered a method to study complex systems thanks to the possibility that they give to handle complex models. Under this extent computer simulations are also considered as virtual experimental laboratories to study phenomena that are difficult to observe directly. However there are also many objections to the use of simulations in science. Daniel Dennett in [2] asks if we can consider real motion what we see in the Cellular Automata world or if it is only apparent motion (the consideration can be referred to Multi-Agent based Simulations as well). Dennett says "The flashing pixels on the computer screen are a paradigm case, after all, of what a psychologist would call apparent motion [...] should we say at least that these moving patterns are real?". Dennett, explaining another objection to the use of Cellular Automata simulation in [2] affirms that "There has been a distinct ontological shift as we move between levels [...] whereas at the physical level there is no motion, and only individuals, cells are defined by their fixed spatial location, at design level we have the motion of persisting objects [...]". In fact when a real phenomena is studied by means of software entities that interact in a computer, it is not easy to demonstrate the correspondence of the dynamics observed in the computer to the ones that belong to a real phenomena. These are not the only objections to the "scientific" use of simulations. The introduction in a simulation of a certain amount of arbitrary details (or assumptions), that are not derived from observations, is often necessary in order to make the simulation run. This makes then difficult to understand which outputs of the simulation are really meaningful and which are instead effects of the introduction of the arbitrary details that we put into the simulation. Moreover, simulations are also criticized to be too simplified in respect to reality as it does not exist a defined criteria to guess if simplifications operated in building a simulation are the good ones. Other considerations regard the fact that simulations do not tell anything new as they are fully deterministic and just give back as output the same information that we put in input. Although many of these objections can be directed also to common scientific practice (more considerations about this topic can be found summarized in [3] [4]) it is undeniable that scientific investigation by simulation is an activity that has to be performed carefully; in fact the possibility of misinterpretation of data is higher than in common experimental practice, exactly due to the "ontological shift" mentioned by Dennett. Winsberg in [1] makes clear that many layers of models are involved in a simulation and different resources are used in each inferential step that is presumed in the shift from one layer to another. Winsberg, again in [1], explains that by these steps the simulation, from an existing theoretical knowledge, attempts to extract new knowledge about the system being simulated. This article explores the role of a specific kind of computer simulations, the MABS, in scientific investigation. In Section II-A is presented a brief overview of some definition of model in science and the relation that exists between a model and a theory. In Section II-B is introduced the scientific cycle in experimental science, in order to make, in Section III, a correlation between models in science and models in simulations. The point of view of the article is that computer simulations are not only scientific “models” of phenomena but are constituted of different layers of abstractions and present one step more in relation to scientific abstractions in experimental science. A schema of the abstractions that are implicitly involved in simulation practice is proposed and, in Section IV, we suggest that, if this activity is interpreted under this perspective, an answer can be found to some of the objections that are often moved to the use of simulation in scientific investigation. In Section V are stated the conclusions.

II. Models and Theories

A. A reflection on Models and Theories

Doran and Gilbert in [5] define a model as something that is similar to a target system *T* but easier to observe. A model *M* of a target system *T* consist in an entity that becomes the object of study in place of *T* in all the cases where this last cannot be studied or observed directly. Doran and Gilbert explain that the idea of the possibility of studying a model of *T* in place of *T* is based upon the conviction that if something is proved to be true in the model then it must be true also in the target system if, in the design of the model, some characteristics of the behavior of the target system have been “captured” by the model. This definition of model is close to the one given by Bruce Edmonds in [6]. Edmonds defines a model as something that “enables an inference process such that the process enabled in the model corresponds to some aspects of an observed process”. For this reason the result of the “inferential process” in the model, if the initial conditions are set properly, remarks the author, predicts some aspects of a subsequent state of the system that is under study. According to Edmonds’ opinion, in science a natural process is *encoded* in a model; the model performs the *inference process* and then the results of the inference are *decoded* and projected to reality in order to state a *prediction*. Edmonds in his article adds that something is a model of “something else” if the diagram in Figure 1 “commutes” and though same results follow both the lines. Under this perspective the *inference process* is thus assigned to the model. Analogously R.I.J. Huges in [7] explains that a model must be analyzed by three activities of mind that are: *denotation*, *demonstration* and *interpretation*. In Huges’ opinion, *denotation* is the “core” of the representation and consist in symbols that stand or refer to parts of the target system; *demonstration* refers to the internal dynamics of the representation (the model) whose effects can be examined; *interpretation* instead consist in the examination of the behavior of the model in order to draw conclusions on the behavior of the world. Eric Winsberg in [8] states that as “models are partially independent of both theories and the world [...] they can be used as instruments of exploration in both domains”. But for this same reason, that a model is independent both of empirical facts and of theories, the translation of the model in the target system must be accurately specified. This independent status of models is remarked also in many other works found in literature (see [9] [10]).

We have spoken about the role of models in relation to theories. But what is then a scientific theory? M.L.Dalla Chiara and G.Toraldo di Francia in [11] explain that a theory is a form of knowledge that resumes a set of laws that can be applied to infinite different cases. The authors add that a form of knowledge has an axiomatic structure and from that axioms, or initial postulates, by a relation of logical consequence is established the set of derived propositions that the theory asserts. This is what is called syntactic definition of theory, but it does exists also a semantic definition stating that some kinds of theories are isomorphic to reality and that the inference steps, bringing from the set of assumptions to the consequences of the theory, are not deductive (see Winsberg [1]).

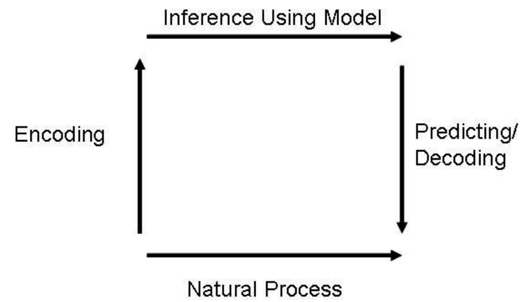

Fig. 1. Edmonds’schema representing the role of model in science

From these considerations we can infer that a model in science derives from a theory, but it is also something that corresponds to reality, although it is distinct also from it. That “correspondence”, to be taken for granted, must be made explicit although it can be proved only by empirical facts.

B. Model, Theories and Phases in Physics

Experimental science is founded on the concept of experiment and of observation of phenomena. David Hestenes in [12] asserts that “the construction of a physical model reduces a real system to an abstract model that it is possible to translate in a mathematical form”. Under his perspective a “mathematical model is at a higher abstract level and constitutes the highest abstraction of the knowledge process” (i.e. equations). The description of the observed phenomena is then constituted by the analytical or the graphic relations between sizes [12]. Axel Gelfert, in [13] states that a mathematical model, in the sense close to the one meant by Hestenes, is a set of equations, theorems and definitions but, Gelfert remarks, the equations do not constitute, by their own, a model of anything (neither of a class of phenomena) unless an interpretation that connects the variables with aspects of observable phenomena is established. We exemplify this process with the schema in Figure 2. In the same direction is also the definition given by Dalla Chiara and Toraldo di Francia that in [11] claim that a physical model can be identified with a structure $M = <Mat, Exp, Tra>$ where Mat represent the mathematical part, Exp represents the experimental part, and Tra is a function of translation that associates a mathematical interpretation to the elements of the experimental part. Axel Gelfert again in [13] underlines that mathematical models, like other kinds of models, require background assumptions for their interpretations. This

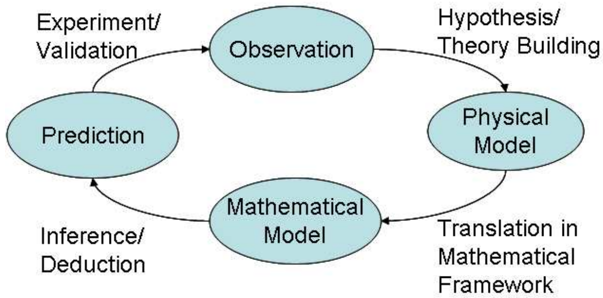

Fig. 2. The schema of the phases in physics.

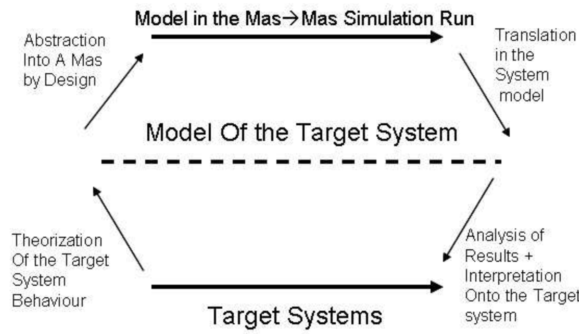

Fig. 3. Edmonds' schema revisited by the means of the introduction of another intermediate level.

perspective is not contrary either to what says Huges that in [7] makes clear that physical theories are not statements about physical world, on the contrary, they are statements about *theoretical constructs* and only if the theory is satisfactory then these constructs stand in a particular relation to the world. In next section some considerations about MABS and models are introduced.

The role of the mathematical model in physics is clear, and it appears to us in tune with the considerations about models that we saw in the previous paragraph.

III. Models and Phases of a Simulation

Our interpretation is that Multi-Agent based Simulations match many of the definitions of model that have been exposed in the precedent paragraphs. Edmonds in [6] states that MABS attempt to model Multi-Actor Systems with a Multi-Agent System. The attempt is to investigate a system by the construction of a MAS model and the analysis of the behavior of the MAS when it runs. In the opinion of Edmonds what distinguishes MABS from other forms of modeling is that simulations based on a Multi-Agent System adopt a MAS as formal model (in physics that role is of mathematics). Often in MABS object-actors or other entities of the system under study (the target system) are mapped onto agents in the MAS. Edmonds explains that the entities in the real system correspond to those of the agents in the MAS, while interactions between entities are correspondent to interactions rules between agents. Edmonds considers a MABS as a model of the target system. In his work he does remain at a high level and suggests interesting epistemological considerations about the MABS. On the basis of his work we would like to attempt a step further and try to detail the abstractions that constitute a MABS. A first consideration is that in a MABS reality is not directly mapped in a MAS, as one might be induced to think by looking casually at Edmonds' diagram, but what that is mapped it is a model of reality (see Figure 3). This model is an intermediate step between reality and MAS. Looking deeper at the different passages implied in the construction of a simulation model using a MAS, it appears clear that implicitly we are working also with other intermediate models. Each model represents a level of abstraction. In fact when we want to study a phenomenon by a computer simulation a first theoretical representation has to be translated through many others steps before it can run on a computer. For this reason to use computer simulation to do science, it must be kept a trace of all the passages that from reality bring to the software code.

Our proposal is schematized in Figure 4. In the left side of the schema are shown the existent levels that we identify between reality and software code. The circularity of the process of simulation described by Edmonds is maintained in our proposal. In the right side of the schema in Figure 4, in fact, are described the necessary steps that have to be followed to turn back simulation results to reality. In the next part of the paper we describe in details the meaning and the importance of each level shown in the schema. Edmonds in his article identifies also some phases in the simulation process (*Design*, *Inference*, *Analysis* and *Interpretation*) and we will see how they can be mapped in our schema at the end of next section. Now we will describe in details the meaning and the importance of each level shown in the schema.

A. Levels of Abstraction in simulation Building

In this section we give a brief explanation of each of the levels shown in the left side of our schema. The levels described below have to be considered levels of abstraction implied (although sometimes implicitly) in MABS. It is not in the scope of this work to give engineering guidelines to translate the requirements of a system into the software implementation, but we want to give a conceptual map of the structure of the practice of simulation.

1) **Target System**: this level of abstraction is constituted by the object of study that is determined by a specific point of view on a portion of reality, considered "isolated" from the rest of the universe. This first abstraction is the result of the identification of the problem that we want to examine and it is achieved by the Phase 1 (*Observation* in our schema). For example, the target system could be the urban road system if the goal of the simulation is to study the urban traffic problem. An example of a Multi-Agent based traffic model can be found in [14].
2) **Abstract Model**: the second level is the abstract model of the target system. The model construction (Phase 2) consists in the definition of the elements that are

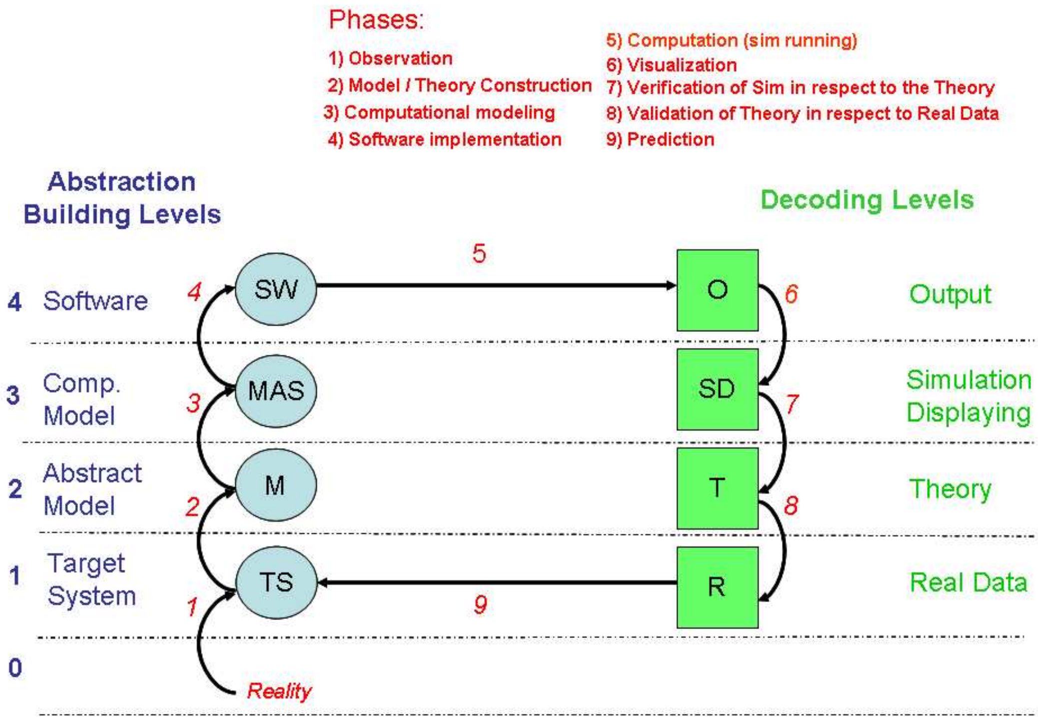

Fig. 4. The schema shows the levels and phases involved in a simulation process. Notice that, with except of reality that has to be considered extra-model, the more abstract levels are found in the bottom part of the diagram.

considered constitutive of the Target System, and in the formulation of a set of hypothesis (conjectures that constitute our intuitive theory) about which rules govern their behaviors and interactions. This definition, at the beginning, can be also not rigorous. For example, in the case of the traffic, our abstract model is constituted by the elements of the environment (i.e. roads, crossings, traffic lights etc.), the individuals that populates the environment (cars, motorcycles etc.), the rules of behavior in this context (vehicles go along the road in one direction, they stop when the traffic light is red, they keep a security distance, etc.) and the properties of the elements that have been identified in that context (each vehicle has a speed etc.).

3) **Computational Model**: the third level is the specific computational model that has been adopted to represent the abstract model (see Phase 3 in the schema). The computational model is always a formal model (it can be a specific MAS). At this level the elements and the rules described in the precedent level are formally defined using proprieties and concepts of the MAS model (i.e. agents, signals etc). In the traffic example cars and traffic lights can be represented, in a MAS, by different kinds of agents, while the behavior of the single agents is modeled giving to the agents the possibility to emit and interpret specific signals (the red stop lights can be represented by a signal sent to car that is following).

4) **The software model**: the fourth level is the software translation of the Computational Model (Phase 4). The computational model, as it is, is an abstract definition and must be translated in a specific program language.

At the end of this 4 steps the original target system will be completely translated in something that is ready to be computed by a machine. The translation of that model into another implies the "encoding" of a model in the language of another. This is an activity that is intrinsic to the MABS and it is strictly related both to scientific activity and to technological constrains. It is important to be conscious of all these passages because each translation can introduce errors and sometimes also a lack of information due to the necessity to use a less expressive language (e.g. the description of the abstract model in the language of the MAS model). Therefore, each abstraction represents another step far from reality because it implies a transformation of the first model into something else. For this reason a "conversion key" that establishes what all the elements of the various models represents, is needed in order to maintain a correspondence with reality and use simulation for scientific investigation.

B. Levels of Decoding

The simulation process proceeds back in the direction of reality by steps that go through several abstraction levels and that help to check the effects of the assumptions of the previous phases and eventually to detect errors or limits of the adopted

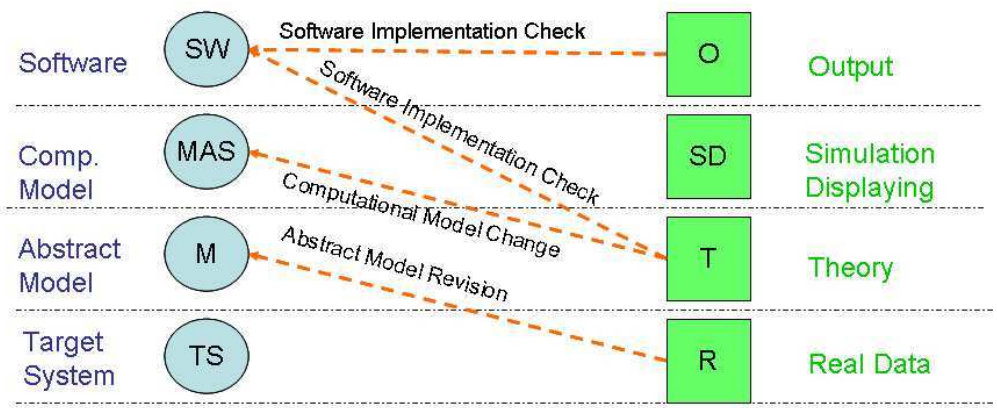

Fig. 5. This schema shows the levels involved in revision phases. of a simulation.

frameworks (e.g the selected MAS). Where it is necessary, in fact, may be necessary to come back to the previous levels and revise one or more of the models involved in the simulation. This operation allows also to correct mistakes introduced in translation phases, and to modify some models by the introduction of new elements that have been demonstrated to be necessary in order to obtain meaningful results (see Figure 5). The four decoding steps are presented below:

1) **Output**: outputs are the simple results of the software execution (Phase 5). If some errors are detected in this step it is necessary go back to the software code.
2) **Simulation Displaying**: the envisioning of the outputs (Phase 6) is often the only way to operate with the simulation dynamic data, because simple outputs are usable only at a machine level. This step is not trivial because consists in the representation of real entities by objects that do not necessarily offer a realistic visualization of the phenomena. The aim of Simulation Displaying in fact is to give an immediate and readable interpretation of the Outputs. This passage is often another abstraction jump that needs a translation key. The importance of the visualization of simulation output data has been largely discussed in many articles (i.e. see Batty and Smith in [15]). This step is very important, in particular for MAS-based simulations of complex systems, because the direct observation of the dynamics envisioned, allows to detect unexpected behaviors. These anomalies will be resolved in the next steps, and they could induce to a revision of the software, of the MAS, or the of Abstract Model. If we focus on the MAS-based traffic model example, at this level we can observe on the screen the dynamic evolution of the simulation constituted by the virtual cars that go along the roads.
3) **Theory**: after the visualization, the correctness of the simulation displaying data must "verified" (Phase 7); in other words it is necessary to check if our initial assumptions at the abstract level are captured by the simulation. This phase is not aimed at evaluating if the assumptions of the theory are correct, but only at verifying if these assumptions are maintained in the simulation. During this phase, we may need to go back to the software or to the MAS to correct mistakes and revise our computational modeling choices if we have some bad feedbacks from the displaying of the simulation. To turn back to traffic example, thanks to visualization phase we can observe the cars that move, brake, form a queue and so on. For example we could notice an anomalous behavior near crosses because cars do not give way correctly as we have assumed in our theory. Therefore we must go back to software in order to check if we did mistakes in the implementation or we may be forced to check if we have wrongly mapped the rules of the Abstract Model in the language of the computational model.
4) **Real Data**: after the verification phase next step is validation phase (Phase 8). If in the previous step we have decided that our theory is well represented by the simulation, now we can validate the theory in relation to real data collected in the target system. In this phase the results of the simulation must be related to real and measurable aspects of reality. In a study on the traffic problem, for example, a portion of a real street must be simulated and the verified results must be compared to the real data collected by observations of reality. If after this check simulation results are considered not reliable, it is necessary to turn back to the Abstract Model and change some of the initial assumptions, or it may be decided the introduction of new elements that at first time were wrongly considered not relevant for the representation of the Target System. The not realistic formation of very long queues, for example, could be resolved with a revision of the interaction rules between the cars and by the introduction of the possibility to overtake when the road on the other way is free.

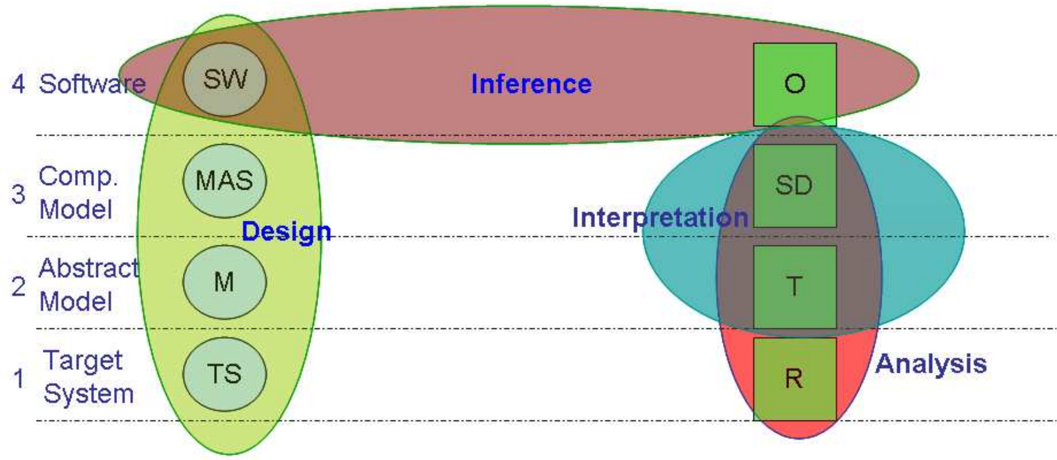

Fig. 6. The figure shows how classical phases of simulation process (as can be found in Edmonds' work) are mapped onto our schema

If the theory is validated by available data then it is possible to make predictions for future states of the Target System (Phase 9).

In reference to the work of Edmonds [6] the phases of the simulation process are identified as *Design*, *Inference*, *Analysis* and *Interpretation*. These phases in our schema are visible in Figure 6. The phase of Design in our schema involves all the 4 abstractions, as to say the Target System, the Abstract Model, the Computational Model and the Software. The Inference phase of Edmonds belongs to the level of the Software abstraction of our schema, the phase of Analysis involves from the level of Displaying to Real Data, while Interpretation regards Simulation Displaying and the Theory.

IV. Some results for the simulation practice

In this section the introduction of an example will clarify our considerations about the aid that the conceptualization given in the proposed schema can give to simulation activity. Therefore we will now introduce the work described in [16]. In particular Balmer and Nagel describe a MAS simulation focused on traffic roundabouts which is applied to a specific Zurich area. During the phase of Design the authors decided to represent cars as particles that move along tunnels (driving routes) representing streets. The spatial model is continuous (the position of each particle is determined by a pair of coordinates) and the particles can be considered agents because they have specific behavior rule (agents must respect the physical rules of acceleration, they have a specific desired speed, they must respect other agents in the system decelerating or overtaking if a slower agent drive in front of them, etc.). Agents also hold information about their destination. Therefore the target system is also in this case the traffic dynamics, the abstract model is constituted by the assumptions and rules like the ones presented above, while the chosen computational model is constituted by a set of physical equations describing the particles motion. During what we have called "the analysis", in particular in the phase that in our schema is identified with the number 8 (validation), a problem was detected: in some situation the cars were subjected to forces exerted in opposite direction (the forces implied were the one directed towards the desired way and the repulsion force aimed at the avoidance of collisions with other cars) and they became unable to move. This problem could not be identified in the design process because it can be considered as an emergent phenomenon that occurs only in some situations and in some specific spatial contexts. This event is a relevant simulation result, because it can be a hint of a possible congestion. In this framework it is then useful to record it, but since this is not a routinary event, the behavior of agent drivers in this situation cannot modelled in a simple way. The decision in this case could have been to turn back to the abstract model and operate a change (i.e. discretization of space or the introduction of giving way rules). However Balmer and Nigel anyway decided not to operate these big changes of their abstract model, but they opted for a minor modification: the introduction of a compenetration rule in order to force the cars to go beyond the stall situation and avoid the deadlock. This is an abstract and gross representation of a set of complex behaviors that can be assumed by human drivers to solve a stall. Thanks to this solution the global behavior of the cars (in sense of aggregate data) maintains a better similarity with the observed reality. In our opinion Balmer and Nigel could detect the problem and adopt a good solution for a correct and scientific use of simulation because they were aware of all the steps done in the previous design phase. It is fundamental to be aware of all these passages as it would be an error to map directly the real world in the software code. if we follow strictly all the passages shown in the schema, it is possible to keep track of the translations of our initial assumptions and, in this way, we can easily avoid the introduction of arbitrary details whose influence cannot be kept under control, as pointed out by some of the critics to simulation practice that we summarized in Section I.

We then suggested that simulations can be used as a valid tools for scientific investigation as the cycle of a simulation

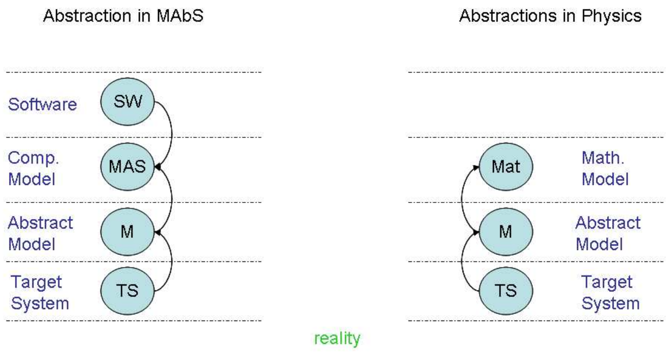

Fig. 7. The schema in figure shows on the left side the abstractions level implied in a simulation in comparison to those (on the right side) implied by experimental science.

reflects the cycle of a science like physics, although a further abstraction is involved in simulation as it is shown in Figure 7. If we come back to reality following the proposed schema in all its passages it is possible to obtain meaningful results that are connected with reality, especially if the revision phases are performed accurately. This practice is analogous to the one followed by a science like physics where a first phase of simplification and abstraction (in newtonian physics reality is so simplified that bodies are seen just as simple points without extension, see [12]) follows an experimental phase by which the results obtained in the simplified model are projected back to reality. These considerations can help to give an answer to other of the objections presented at the beginning of the paper.

V. Conclusions

In this paper we began reporting some definitions of what can be considered a model in science and we briefly described the scientific cycle in physics. Then we raised some questions about MABS and their role in scientific investigation. We started from the work of Edmonds to introduce our investigation in the nature of simulations in the attempt to give some answers. Our conclusion is that simulations are not "models" but are a compound of many models at different levels of abstractions. We pointed out that in the encoding or decoding of one level into another is situated the risk to loose the scientific purpose of simulations, if a translation criteria between models is not stated explicitly. When one modeler is led to revise his/her theory about the simulated domain after an analysis of simulation results and their validation, simulation can be considered a useful tool for scientific investigation. On the other hand, if the simulation has been deeply tested and we can trust simulation results, we focus on the phase of simulation prediction. In this last case we are working with a tested instrument and we are not using MABS to do scientific investigation but simply, for example, to help decision makers. It must be noted that other areas of Computer Science and engineering deal with models and abstractions of real systems, in particular Software Engineering. For example the problem of correspondence between models is considered also by the Model Driven Architecture Approach (MDA), that focuses on the importance of the mapping between the models present at different levels of Software Engineering practice. One of the purpose of MDA is to give guidelines for integration of Information Technologies in order to assure interoperability between systems (for an introduction and first references on MDA see for example [17]). Another meaningful example is the work of M. Jackson [18] that analyzes under the perspective of Problem Frames (PF) the relation that, in Software Engineering, exists between the requirements, the real world, and the machine considered as the general purpose computer that will execute the software. MABS share several aspects with Software Engineering, because the proposed schema reminds several iterative software development processes that are used also to project and build an effective simulation tool. MDA or PF definitions take in consideration a class of problems that is involved also in MABS, but our study does not take into consideration deeply the technical problems that are strictly related to the machine, the software and the specific available technologies. The aim of this work is in fact to state some general considerations that could help to preserve scientific contents through the many models involved in MAS simulation practice.

References

[1] E. Winsberg, "Sanctioning models: epistemology of simulation", Science in Context, 12, 275-292, 1999.
[2] D.C. Dennett, "Real Patterns", The Journal of Philosophy, Vol. 88, No. 01, pp. 27-51, 1991.
[3] K. Richardson, "The Problematisation of Existence: Towards a Philosophy of Complexity", In proceedings of ISCE Workshop on Complexity and Philosophy, 2002.
[4] D. Parisi, "Simulazioni", Il Mulino Eds, Bologna, 2001.
[5] J. Doran and N. Gilbert, "Simulating Societies: an introduction", in N.Gilbert e J.Doran Eds, Simulating Societies: the Computer Simulation of Social Phenomena, UCL Press, pp. 1-18, London, 1994.
[6] B. Edmonds, "The Use of Models - Making MABS More Informative", Lecture Notes in Computer Science, Page 15, 1979, Jan 2000, .
[7] R.I.G. Hughes, "Models and Representation", Philosophy of Science, pp. S325-S336, 1997.
[8] E. Winsberg, "Simulated Experiments: Methodology for a virtual World", Philosophy of Science, pp. 105-125, 2003.
[9] M. Morrison and M.S. Morgan, "Models as Mediators, Perspectives on Natural and Social Science", Cambridge University Press, 1999.
[10] N. Cartright in "the dappled world. A study of the boundaries of Science", Cambridge, Cambridge UP 1999.
[11] M.L. Dalla Chiara and G. Toraldo di Francia, "Introduzione alla filosofia della scienza", Laterza, Roma-Bari, 1999.
[12] D. Hestenes, "Modeling Games in the Newtonian World", Am. J. Phys. pp. 732-748, (1992).
[13] A. Gelfert, "Simulating Many-Body Models in Physics: Rigorous Results, 'Benchmarks', and Cross-Model Justification", in Proceedings Models and Simulations, London, 2006.
[14] K. Dresner and P. Stone, "Multiagent Traffic Management: A Reservation-Based Intersection Control Mechanism", In Proceedings of the Third International Joint Conference on Autonomous Agents and MultiAgent Systems, p. 530–537, 2004.
[15] M. Batty and A. Smith, "Urban Simulacra: From Real to Virtual Cities and Back and Beyond", Architectural Design, Sensing the 21st Century City: The Net City Close-up and Remote, Eds. David Grahame Shane and Brian McGrath, 2005.
[16] M. Balmer and K. Nagel, "Shape Morphing of Intersection Layout Using Curb Side Oriented Driver Simulation", in Innovations in Design & Decision Support Systems in Architecture and Urban Planning, Springer–Verlag, p. 167–183, 2006.
[17] A. Wegmann and O. Preiss, "Strengthening MDA by Drawing from the Living System Theory", in WiSME@UML 2002, Workshop in Software Model Engineering, Dresden, Germany, 2002.
[18] M. Jackson, "Problem Frames and Software Engineering", Information and Software Technology 47, p. 903–912, 2005.