Chains of Reference in Computer Simulations

Franck Varenne

fondation
maison des
sciences
de l'homme
Programmes
scientifiques

GeWoPs
Working papers du
GEMASS
Groupe d'étude
des méthodes de
l'Analyse sociologique
de la Sorbonne

Chains of Reference in Computer Simulations

Franck Varenne

N°51 | Octobre 2013

This paper proposes an extensionalist analysis of computer simulations (CSs). It puts the emphasis not on languages nor on models, but more precisely on symbols, on their extensions, and on their various ways of referring. It shows that *chains of reference* of symbols in CSs are multiple and of different kinds. As they are distinct and diverse, these chains enable different kinds of remoteness of reference and different kinds of validation for CSs. (...)

Working Papers Series

Fondation Maison des sciences de l'homme - 190 avenue de France - 75013 Paris - France
http://www.msh-paris.fr - FMSH-WP-2013-51

Chains of Reference in Computer Simulations

Franck Varenne

Octobre 2013

The author

University of Rouen & GEMASS (CNRS / Paris Sorbonne)

franck.varenne@univ-rouen.fr

The text

This research was lead in the framework of the ANR Program TRANSMONDYN 2011-2014 (dir. Léna Sanders). To be published in S. Vaienti, P. Livet (eds.), *Simulations and Networks*, Aix-Marseille, Presses de l'Université d'Aix-Marseille, coll. IMéRA, 2013.

First version: 2011/09/20 - Modified version: 2013/04/06.

This paper is published by the Groupe d'étude des méthodes de l'Analyse sociologique de la Sorbonne (GEMASS - CNRS, Paris Sorbonne, FMSH) in Paris, under the cover of the Fondation Maison des sciences de l'homme's collection of working papers.

Citing this document

Franck Varenne, *Chains of Reference in Computer Simulations*, FMSH-WP-2013-51, GeWoP-4, octobre 2013.

Informations et soumission des textes :

wpfmsh@msh-paris.fr

Fondation Maison des sciences de l'homme
190-196 avenue de France
75013 Paris - France

http://www.msh-paris.fr
http://halshs.archives-ouvertes.fr/FMSH-WP
http://wpfmsh.hypotheses.org

Abstract

This paper proposes an extensionalist analysis of computer simulations (CSs). It puts the emphasis not on languages nor on models, but more precisely on symbols, on their extensions, and on their various ways of referring. It shows that *chains of reference* of symbols in CSs are multiple and of different kinds. As they are distinct and diverse, these chains enable different kinds of remoteness of reference and different kinds of validation for CSs. Although some methodological papers have already but implicitly taken into account the heterogeneity and variety of the relationships of reference in CSs, hence of cross-validations, this diversity is still overlooked in the epistemological literature on CSs. As a consequence, a particular outcome of this analytical and explicitly extensional view is an ability to classify existing epistemological theses on the epistemic status of CSs according to what their authors choose to select and put at the forefront: either the extensions of symbols, or the symbol-types, or the symbol-*tokens*, or the *internal denotational hierarchies* of the CS seen as a whole or the references of these hierarchies to *external denotational hierarchies* seen as wholes. Through the adoption of this extensionalist view together with its precise conceptual differentiations, it also becomes possible to explain more precisely the reasons why some complete reduction of the epistemic role of CSs to classical epistemic paradigms such as "experience", "experiment", or "theoretical argument" remains doubtful. On this last point, in particular, this paper is in agreement with what many epistemologists already have acknowledged. But it proposes new conceptual means - new in this context - to explain the situation further.

Keywords

computer simulation, numerical simulation, agent-based simulation, epistemology, reference, chains of reference, extensionalism, computation, epistemic status of simulations, denotational hierarchy, inscriptionalism

Les chaînes de la référence dans les simulations informatiques

Résumé

Cet article introduit une *analyse extensionnaliste* des simulations informatiques. À cette fin, il ne se focalise ni sur les langages ni sur les modèles mais, plus finement, sur les symboles, sur leurs extensions et sur leurs différents modes de référence. Il montre que les *chaînes de la référence* des symboles intervenant dans des simulations informatiques sont multiples et de différents types. Comme ces chaînes sont diverses et distinctes, elles autorisent différents types de distance - ou éloignement, *remoteness* selon N. Goodman - entre symboles et référence, et, par là, différents types de validation pour les simulations informatiques. Bien que certains articles méthodologiques aient déjà implicitement pris en compte le rôle de ces relations hétérogènes et variées entre les symboles et leurs références ainsi que le rôle des validations croisées dans les simulations informatiques, cette diversité reste négligée dans la littérature épistémologique. Une des conséquences de l'approche analytique explicitement extensionnaliste que nous proposons d'introduire ici est dans un premier temps le développement d'une capacité à classer les thèses épistémologiques existantes au sujet du statut épistémique des simulations selon ce que leurs auteurs choisissent de mettre en avant, à savoir notamment : tantôt les *extensions* des symboles elles-mêmes, tantôt les *types* de symboles, tantôt les tokens de symboles tantôt les *hiérarchies dénotationnelles internes* à la simulation prise comme un tout, ou les références de ces hiérarchies à des *hiérarchies dénotationnelles externes* prises comme des touts. Dans un second temps, en adoptant explicitement une telle conception extensionnaliste ainsi que ses distinctions conceptuelles précises, il devient également possible d'expliquer les raisons pour lesquelles il reste douteux d'espérer que l'on puisse réduire tout uniment le rôle épistémique des simulations informatiques aux paradigmes épistémiques classiques comme « l'expérience », « l'expérimentation » ou encore « l'argument théorique ». Sur ce dernier point, en particulier, cet article est en accord avec ce que maints épistémologues des simulations ont déjà reconnu. Mais il propose des moyens conceptuels nouveaux (nouveaux dans ce contexte) pour expliquer davantage les facteurs qui sont à l'origine de cette situation.

Mots-clefs

simulation informatique, simulation numérique, simulation à base d'agents, épistémologie, référence, chaînes de la référence, extensionnalisme, computation, statut épistémique des simulations, hiérarchie dénotationnelle, inscriptionalisme

Content

Introduction

There still exists no general agreement on the epistemic status of computer simulations (CSs) in science; it seems doubtful that there ought to be one. Both the scientific and the epistemological literatures show a huge diversity of practices and associated epistemic functions for CSs[1]. Furthermore, the recent spreading of complex computer-aided computations of simulation models in disciplines traditionally distinct from mathematically grounded ones[2] such as developmental biology, human geography, computational sociology or computational archeology, has reactivated the quite old, but specific debate on the empiricity of CSs[3].

In front of this pervasive diversity of practices and meanings of CSs, a *prima facie* strategy for the epistemologist could be to adopt a culturalist standpoint. Many social and historical studies of contemporary sciences using models and simulations have convincingly shown that idiosyncratic material cultures, social dynamics and philosophical preferences operating in various research programs can, to some extent, explain this diversity[4].

Our approach is based on the idea that this diversity does *not only* rely on such idiosyncrasies. It is based on the hypothesis that it could be worth letting the conceptual analysis of the nature and status of individual symbols at stake in each CS go a bit further. For instance, it could be worth going further than simply assuming that the diversity of viewpoints and interpretations of complex simulations is a matter of languages, of cultures or even of worldviews. Positive hints in favor of this working hypothesis towards a more scrutinized analysis of *symbols* (and not only of "languages" which is much more restrictive) emerge from the fact that the *computational turn* occurring in sciences since the 90s has led simulation practitioners, especially those working in computational life sciences and computational social sciences, to admit that not all computer simulations can meaningfully be reduced to some numerical simulations of an explicit pregiven mathematical model of their target system. That is: even if practitioners still call them *simulations*, computer-aided computations of complex models of simulation appears to them as putting at the forefront a *new computer-aided mode of operating on symbols*, significantly different in their referential commitments from the ones used by traditional computer-aided numerical simulations. For instance, in the domain of computer simulation for economics, practitioners noticeably emphasize the conceptual differences between the traditional simulations of mathematical models firstly introduced by Jay W. Forrester in the 1960s[5] and the agent-based simulations currently developed in ACE (Agent-based Computational Economics)[6]. In the domain of agent-based social simulation too, the difference is particularly underlined[7]. In fact, since the spreading of object-oriented programming and object-oriented modeling techniques[8], it appears that some conceptual distinctions clearly have to be made between at least three different types of computer simulations according to the diverse nature and meaning of their *mode of operation on symbols*: 1. *model-driven simulations* (numerical simulation), 2. *rule-driven simulations* (algorithmic simulations) and 3. *object-driven simulations* (software-based simulation)[9].

Our general method in this paper will consist first in substantiating more precisely the distinctive elements between these three avowed types of CS. Our auxiliary working hypothesis is that this first substantiation will help us to uncover more largely how the distinctive levels of symbols can diversely operate in a computer simulation's function of referring.

In order to apply this method with the desired precision, we will choose to adopt an extensionalist approach of symbols. The content of this approach will briefly be recalled and explained in

[1] For the epistemological literature only, see *e.g.* Humphreys (1990), Rohrlich (1990), Hartmann (1995), Dietrich (1996), Galison (1996, 1997), Bedau (1998), Winsberg (1999), Stöckler (2000), Di Paolo *et al.* (2000), Varenne (2001), Peck (2004), Livet (2007), Winsberg (2009), Morrison (2009), David *et al.* (2010), Peschard (forthcoming).

[2] Like mathematical physics, chemistry, engineering sciences, mathematical economics or molecular biology.

[3] For this debate, see: Sugden (2002), Humphreys (2004), Peck (2004), David et al. (2005), Mäki (2005), Morgan (2005), Winsberg (2009), Morrison (2009), Phan & Varenne (2010), Reiss (2011), Peschard (forthcoming).

[4] For this approach, see Hacking (1983), Galison (1996, 1997), Keller (2002a, 2002b), Küppers et al. (2006).

[5] Forrester (1968).

[6] See *e.g.* Tesfatsion (2002).

[7] Gilbert (2007), Varenne (2010).

[8] Hill (1996), Hill & Coquillard (1997).

[9] Varenne (2007), Phan & Varenne (2010).

the first sections. The very reason why we choose this approach so as to cope with our specific question on simulations won't be given *a priori*, *i.e.* in the beginning of the paper. First, because it would be out of its scope to begin with an overall justification of such a choice as it largely depends on a whole theory of symbols, depictions and descriptions. Second, because this choice will be shown fruitful at the end of the paper, in its last sections, particularly in its ability to conceptually discriminate between current and sometimes contradictory interpretations of computer simulations as they appear in the epistemological literature on CSs. Third, because, following a remark of Goodman (1977) on this point, an extensionalist approach involves a poor ontological commitment. So, although such a claim may seem disputable or counterintuitive at first glance, an extensionalist approach of the function of referring for symbols is *not* directly contradictory with ontologically more dispendious systems like intensionalist or platonician ones. And thanks to this property of ontological minimalism, an extensionalist approach can play the role of a minimal base of agreement between different systems of philosophy of language and philosophy of knowledge; at least as far as an inquiry on the epistemic status of computer simulation is concerned.

Extensionalism on computations

Computation, symbols and reference

The aims of this paragraph are first to present a sketch of the contemporary extensionalist general theory of symbols chiefly due to the analytical works of Goodman, Scheffler and Elgin[10] and second to begin to substantiate the idea that computer simulations could valuably be analyzed and discriminated through some of the analytical tools developed by this theory.

First of all, let us ask the question: does a CS have anything to do with symbols? And if it does, is this relationship essential or only contingent? Briefly said: what is a *computer simulation*? A distinctive characterization will be given in section 2. But, beforehand and following the common knowledge on this matter, it is possible to characterize it roughly as *a kind of computer aided computation*. From the viewpoint of contemporary computer scientists and programmers, a *computation* itself is generally seen as a *kind of calculation*.

But, by claiming this, one reduces a *computation* to only one of its possible use: *computation on numbers*. Comparatively, Colburn (2004:318) is more prudent:

> Computer science is a science concerned with the study of computational processes. A computational process is distinguished from, say, a chemical or electrical process, in that it is studied "in ways that ignore its physical nature" (Hailperin et al. 1999:3).

Hailperin *et al.* (1999) themselves give the following precisions:

> A process is a dynamic succession of events – a happening. When your computer is busy doing something, a process is going on inside it. What differentiates a computational process from some other kind of process (*e.g.*, a chemical process)? Although computing originally referred to doing arithmetic, that isn't the essence of a computational process: For our purpose, a word, for example, enjoys the same status as a number, and looking up the word in a dictionary is as much a computational process as adding numbers (Hailperin et al. 1999:3).

The problem of reducing computation to calculation is often overlooked. It insidiously leads to underestimate the main ideas which originate the conception of Turing-Machines: 1) the role of symbols in general, and not only numbers, and 2) the role of simulation. On this point, it is worth following the argument of Copeland (2004). Going back to the seminal ideas of computer aided computation in Turing works, Copeland shows that because they still are designed as approximate Turing-Machines[11], contemporary computers have to be thought first as *simulating machines*. Hence, they don't have to be reduced to calculating machines.

Here, it seems as if we enter a circular characterization of a CS: a simulation is a computation, which itself is a simulation. In fact, Turing-Machine computations are not conceptually designed to simulate anything: they specifically

[10] Goodman (1968, 1978), Scheffler (1979), Goodman (1981), Elgin (1983), Scheffler (1997).

[11] "Approximate" because, contrary to real Turing-Machines, their memory is non-infinite.

have to simulate a human behavior. More precisely, they have to *emulate* such a behavior. For a given system S, an *emulation* S* of S is another system which *behaves* exactly in the same way (with exactly the same outputs) as the system S, and this for every input and initial parameters. Hence, from a more general and fundamental perspective on contemporary classic computers, a computer first of all has to be characterized as a *machine simulating a human behavior, in the sense of* emulating *it*. But the crucial point here is that not all human behavior can directly be emulated this way, only this *behavior in which human beings manipulate discrete symbols* or, more precisely, *tokens of discrete symbols* (*i.e.* events or instances of discrete symbols). Then, in any of its operations, a classic computer *computes* in that it *emulates* this particular human behavior or task consisting in any decomposable and explicit (*i.e.* non-opaque) *operation on symbols*. From this more general perspective, *a computation is any step-by-step operation on discrete symbols*.

It still is possible to see any operation of this kind as a calculation. But, by doing this, one commits oneself to a disputable reduction. One reduces any discrete symbols to its only ability to refer to numbers (among other things or classes or aspects of things). The generality of the function of referring through these (tokens of) symbols which are at stake in any computer-aided computation is overlooked. It is no wonder that von Neumann (1961) very soon (in 1949) underlined not the fact that digital computers were calculating machines but, more generally, that such approximate Turing-Machines are automata that aim at simulating nothing else than computation in general, just the same way von Neumann supposed computation was physically performed by the human brain[12].

As far as agent-based social simulation is concerned, a similar approach to the one we will suggest here has been proposed by David *et al.* (2005). One of the key claims of this paper is that, in this context, computation is more than an uninterpreted formal calculus. We agree with this claim. But, according to the authors, "the semantic significance of computer programs conveys not only a causal capability, but also an *intentional capability*" (David *et al.* 2005, §3.9)[13]. By "intentional capability" of a social computer simulation the authors mean:

> the recognition that since computation is in one way or another a symbolic phenomenon, or representational, or information-based, or semantical, it is intentional insofar as we assume that the behaviors of computers stand for other things in the world (David et al 2005:§3.10).

The authors equate the property of a computer program to be "a symbolic phenomenon" or to "stand for other things in the world" with the property they call "intentional capability". This is where our approaches split. In the philosophy of mind, to be intentional means more strictly "to be about". This aboutness is said to be a typical property of mental states, for instance; whereas "standing for" is an adequate characterization for "reference" or "symbolization" as recalled by Goodman (1981:121). Then, by directly making these two different properties adequate, the authors remain ambiguous. Do they mean "reference" by "intention"? Or, do they mean "intension" (with an "s", *i.e.* in the sense of "sense" or "meaning") by "intention"? Remaining in this ambiguity, they are in danger of facing the question of the subjectivity of interpretations in social sciences. By identifying symbolization and intentionality, they commit themselves ontologically to using inner representations in their explanations as we can see in the lines preceding their definition of intentional capability:

> The acceptance of a social simulation by a community of observers depends on interpretative aspects that go beyond empirical adequacy, for the semantic significance of computer programs conveys not only a causal capability, but also an intentional capability (*ibid.*).

Even for the sake of science, to admit the very existence of inner representations and of their subjectivity is not in itself a problem. It becomes problematic when we intend to *convoke* and *use* such admitted entities to try to publicly and explicitly (*i.e.* scientifically) explain social phenomena. The problem is that this can lead to the acceptance of possibly incommunicable then incomparable idiosyncratic inner representations and interpretations. The vagueness, opacity and

[12] If we have to agree on the former point, this does not mean that we have to adopt the latter, i.e. the so-called *computationalist* thesis in the philosophy of mind. It suffices to recognize the seminal role of *symbols* in computations, the term being taken in its most general meaning.

[13] Our emphasis.

unreliability of so-called *intensions* are some of the reasons why contemporary philosophers of language have developed an alternative approach of symbols and reference called *extensionalism*.

Here, we briefly characterize first what symbols and references are, according to this approach.

> "Symbol" is used here as a very general and colorless term. It covers letters, words, texts, pictures, diagrams, maps, models, and more, but carries no implication of the oblique or the occult (Goodman, 1976:xi).
>
> "Reference" as I use it is a very general and primitive term, covering all sorts of symbolization, all cases of *standing for* (Goodman 1981:121), his emphasis.

An *extension* of a symbol is all the things or aspects of things (objects, persons) referred to by this symbol. An *intension* is the *idea of the properties all these things are supposed to share* so that they can be told to *belong* to the same extension. If one admits the existence of intensions, it has an ontological consequence on extensions: one has to see extensions as plain and real classes having some kind of real existence. *Intensionalism* is a label which denotes a theory of symbols that assumes the existence of intensions, ideas and/or classes. *Extensionalist* is a label which denotes a theory that refuses them. It denotes a theory of symbols that exclusively focuses on the extensions of symbols, not on their intensions[14]. Such a view refuses to ground any analysis of the functioning of symbols on unclear and disputable entities.

Denotation, null labels and secondary extension

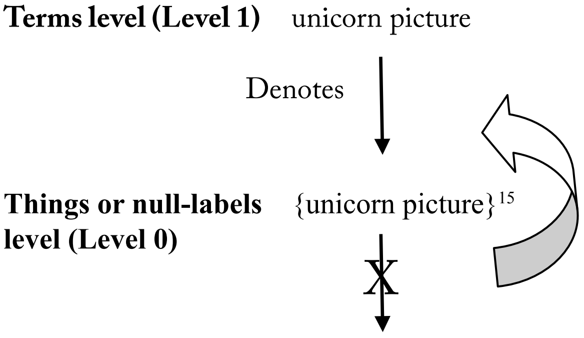
Denotation is "the application of a word or picture or other label to one or many things" (Goodman 1981:121). Naming is the paradigmatic case for referring through denotation. There is a denotation when there is a label. In this section, we will show that, even for an extensionalist approach, there are different ways of referring for symbols and that denotation is only one of them.

It is often objected with respect to extensionalism that symbols exist which refer to nothing. For instance, the term "unicorn" or the term "Pegasus", a unicorn picture or a winged horse picture are symbols with no extension. Such a symbol exists in itself (be it a term, a label, a picture), but its extension is null. Goodman calls them *null-labels*. But the problem is the following: because it refuses intensions, does the extensionalist view entail that we are saying nothing when we speak of a unicorn? Goodman answers:

> When we speak of a picture as depicting a unicorn, even though there are no unicorns to depict, what we are saying, in effect, is rather that the picture is a unicorn-picture; we are saying not that the picture denotes anything, but rather that it is denoted by the term "unicorn-picture" (Goodman 1981:125).

As our schematic representation shows (Figure 1), for this particular mode of reference, the extensionalist answer consists in: first, admitting that null-labels denote nothing but that this is not the proper meaning of our speech, second, adding a symbolic level up (Level 1) to the symbolic level of the null-label (Level 0), third, displacing the *up-down denotation relation* so that it operates not between Level 0 and a hypothetical sub-level of nonentities ,but between Level 1 and Level 0, fourth, admitting that there still exists a non-null extension (referred to by a symbol), because it is the unicorn picture which now serves as an extension as it is denoted by the implicit term "unicorn-picture".

Figure 1 –
Mode of reference for null-labels



Even if it is possible to accept symbols that denote nothing, another objection to extensionalism is that symbols with no extension still seem to have different meanings, hence perhaps different *intensions*. For instance, even though the terms unicorn and centaur denote nothing, few would admit that they mean the same thing. The problem is the following: if one refuses to admit

[14] "Extensionalists are committed to basing interpretations on nothing, but extension", Scheffler (1997:91).

[15] Our convention will be that the curly brackets symbols "{ }" indicate the thing itself, not the symbol denoting the thing.

and hypostatize the intensions of null-labels, as they all have no extension at all, how is it possible to account for their persistent difference in meaning in ordinary discourses? To answer this objection, Goodman suggests considering the parallel compounds of each symbol denoting nothing. A *parallel compound* of a symbol is another symbol formed by adding "picture" or "description" to it (Scheffler 1997:34). For instance, a parallel compound of the term "unicorn" is the term "unicorn-picture". Then, Goodman suggests considering not the extension of the initial symbol, but the one of its parallel compounds. Goodman calls it the *secondary extension* of the initial symbol. As can be seen in the example of the pair "unicorn-picture" and "centaur-picture",

> Terms, in general, have the same meaning if and only if they have the same primary and secondary extensions (Scheffler 1997:34).

Hence, even if "unicorn" and "centaur" have the same extension (null), as "unicorn-picture" and "centaur-picture" have not the same extension, it follows that "unicorn" and "centaur" have different meanings.

Reference via mention-selection

But, a third objection to extensionalism still can arise. For instance, when we say that "a rose is a rose", we mean something like "things are what they are whatever our discourses on them". And, by assuming this meaning, we assume that the first instance (the first word-token) of the term or word-type "rose" has not the same meaning as the second one. Had these two tokens exactly the same meaning, the sentence would have no meaning at all (at least no empirical content). But, it has meaning. Hence, the strategy of the parallel compounds is of no use in this case.

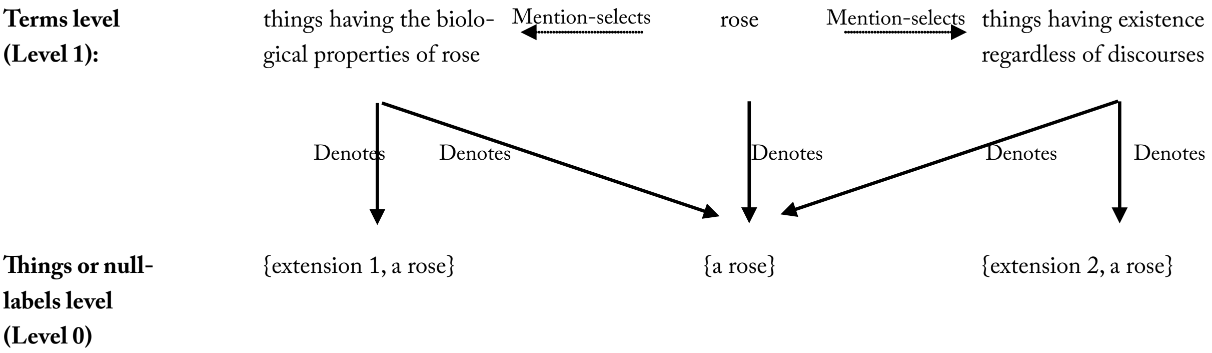
> The notion of parallel compounds implies that they are syntactically distinguishable and assignable to the two tokens differing in meaning. And, the latter condition fails for replicas16 differing in meaning (Scheffler 1997:47).

In order to answer to this new objection, Scheffler notes that we sometimes "apply the term 'man' to select not a man, but a picture" and that "we here apply the term not to what it denotes, but rather to a mention thereof" (Scheffler 1997:48). From this observation, it follows that "man" not only denotes men, but can be applied *mention-selectively* to man-pictures. This functionality of symbols depends on the way these complimentary uses were implicitly taught to us when we were children. Just because we were taught how to use the term "unicorn" by being shown not unicorns, but only pictures or verbal descriptions of unicorns, we simultaneously were taught to use the term 1) as referring to nothing, 2) as referred to by its compounds, 3) but also as mention-selectively referring to pictures and descriptions of unicorns. As a consequence, Scheffler suggests introducing *mention-selection* as another and plain way of referring for symbols.

Beware that a mention-selection is not a denotation. As Quine had underlined in his *Mathematical logic*, "mention" of symbols has to be distinguished from their "use": "we *mention* x by *using* a name of x" (Quine 1981:23). For instance, in the sentence "Boston is populous", the place-name is *used* and the city is *mentioned*. But, in the sentence " 'Boston' is disyllabic", a quotation is *used* and the place name is *mentioned* (*ibid.*). In an extensionalist view on symbols, a denotation is the relationship between a label and its extension. It *selects* this extension. This happens in the case where we *use* the label. But, this label can also *mention* other labels or descriptions known to have the same extension. In this case, the label does select neither its extension nor the extension of its parallel compounds, because it is not *used*. But, it more largely *mentions* some other symbols which *stand at the same symbol level* and which are known to, sometimes, denote the same things. According to this last conceptual distinction, it can be explained that, depending on the context, only some replicas of the word-type "rose" mention-selects certain other terms having a given extension, whereas other replicas of the same word-type "rose" mention-selects certain other terms having another given extension (see Figure 2).

[16] A *replica* of a symbol (or *true-copy*, or copy without fault) is one event of this symbol. It is one given (concrete) instance of this symbol. Replicas do not exist in all systems of symbols. They exist in notational (or at least inscriptional) systems of symbols such as writing. For instance, in alphabetical writing, an event (or token) of a letter has to belong to a *character* (for instance, the character "a") whatever the paper, the pen, the ink, the screen or the police of character this *letter-event* has used. See Goodman (1976:131): "A necessary condition for a notation, then, is *character-indifference* among the instances of each character". Analogously to letters, in alphabetical writings, two distinct *word-tokens* of the same term are replicas of a given *word-type*.

Figure 2 – Reference via mention-selection



Notice that, in this case, we can speak of a *remoteness of reference* in the sense of Goodman (1981). That is: in the sentence "a rose is a rose", the term "rose" is not used; it denotes nothing, but it nevertheless *refers to* something in that it refers (diversely) to some extensions *via* a variety of mention-selection relationships. Hence, even if "rose" is not used in its ability to directly denote a rose, it refers to {extension 1, a rose} *and* to {extension 2, a rose} *via* the function of mention-selection. Here, we have a first example of a *chain of reference*. It is a *two-links chain of reference.*

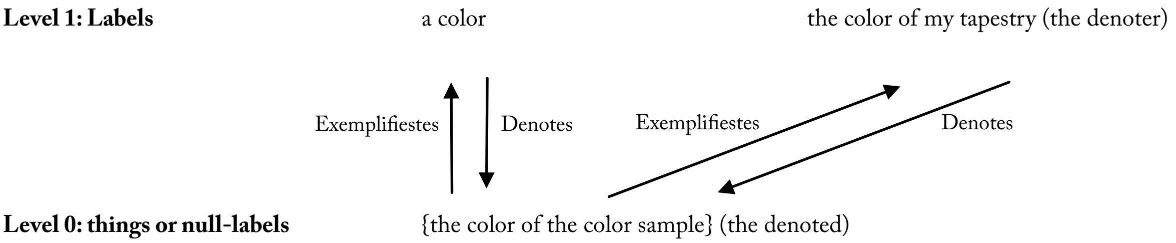
Exemplification

There is another important mode of reference that will be useful for our analysis of computer-aided computations in CSs; this is *exemplification*. A color sample, for instance, does not denote the color of my tapestry. But, it *exemplifies* this color. It exemplifies only the color, but no other feature of my tapestry (its size, texture, etc.).

> Exemplification is reference by a sample to a feature of it (Goodman 1981:124).
>
> Exemplification, then, far from being a variety of denotation, runs in the opposite direction, not from label to what the label applies to, but from something a label applies to back to the label (or the feature associated with that label). Exemplification, indeed, involves denotation, by inversion, yet cannot be equated with the converse of denotation; for exemplification is selective, obtaining only between the symbol and some, but no others of the labels denoting it or properties possessed by it (Goodman 1981:125).

Hence, exemplification is another way of referring. It refers *via* a return reference to a denoter by a denoted (*ibid.*). In our case, the denoter is "the color of my tapestry" and the denoted is the color of the color sample (see Figure 3).

Figure 3 – Reference via exemplification


Level 1: Labels
a color
the color of my tapestry (the denoter)
Exemplifiestes
Denotes
Exemplifiestes
Denotes
Level 0: things or null-labels
{the color of the color sample} (the denoted)

As we will see, symbols occurring and operating in complex CS often use this kind of indirect way of referring. It is important to understand this point if we do not want to reduce *a priori* all the diverse kinds of validation of a CS only to some *structural external validity*[17], *i.e.* to some *isomorphism* between syntactic relationships, between symbols occurring at the same level, within the model of simulation or within the computation, on the one hand, and correlative relationships between field data or models of data, on the other hand. For instance, a term-to-term, but indirect[18] relationship of reference can exist in individual-based simulations in ecology or in *ab initio* molecular dynamic simulations[19]. Of course, *de jure*, this term-to-term relationship is not contradictory with an *ex post* validation thanks to isomorphism detection. But, the point is that the initial conception and the identification procedure of such models of simulation and of their variables are not primarily based on an isomorphic-view, but on the implementation of various individual features diversely exemplified in the symbols that will operate in the computation.

Chains of reference in denotational hierarchies

Three distinct ways of referring are now at our disposal: *denotation* (available even for null-labels), *mention selection* and *exemplification*[20]. Through our precedent explanations and figures, we have seen that, in order to make each of these ways of referring work, symbols have to be developed or represented as belonging to distinct levels. Furthermore, each symbol belongs to a given *system of symbols* which, itself, constitutes a level.

All those levels taken as a whole constitute what Goodman (1981) calls a *denotational hierarchy (DH).* It is a hierarchy in which the direction matters: there is a directional arrangement of these levels. For instance (see our schematic representation in Figure 4), denotation always has to be a relationship of referring going downward, from a label to its labeled, whereas exemplification has to be represented as going in the opposite direction. Whether this direction has to be downward or upward is a matter of representational convention here. The point is that this direction has to be defined once and for all and that it must not change thereafter.

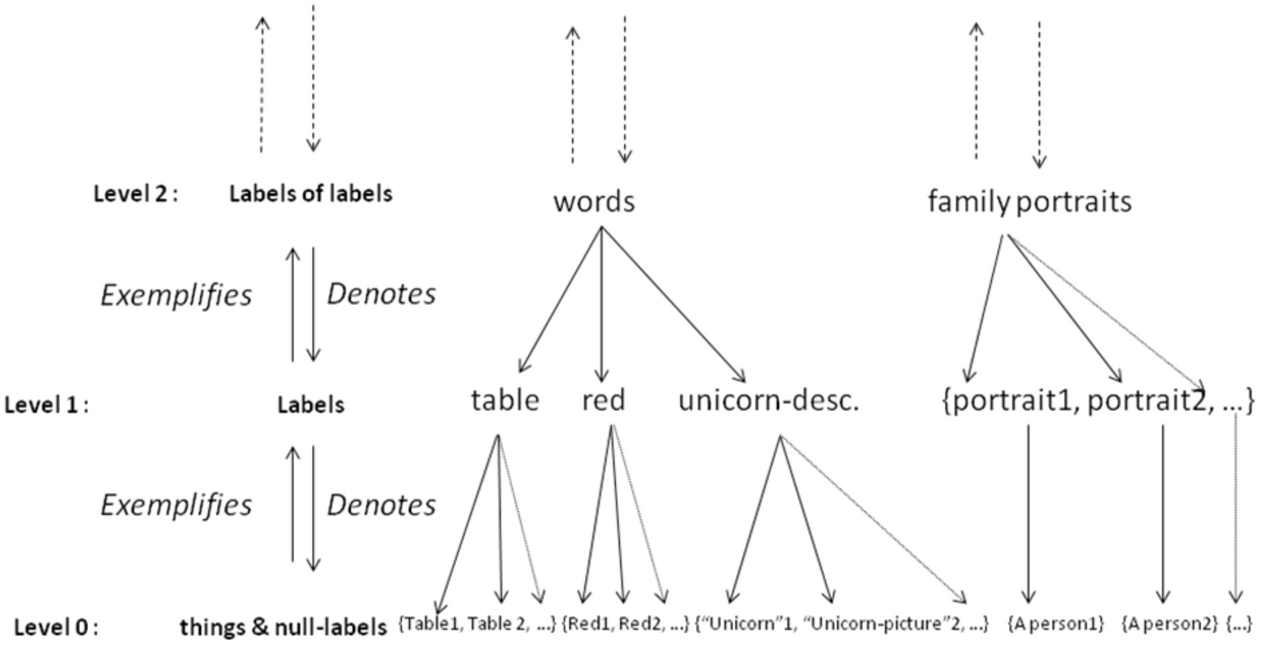
[17] Guala (2003).

[18] This indirectness is due to the remoteness of such a *reference* via exemplification.

[19] See Morrison (2009:45) on this particular claim in the case of molecular dynamics simulations.

[20] There are other ways of referring for symbols such as *expression* and *reenactment*. But we will not use them for our analysis of CSs as they suppose direct reference to human emotional behaviors or gestures. See Goodman (1976), Elgin (1983) and Scheffler (1997).

Figure 4 - A denotational hierarchy



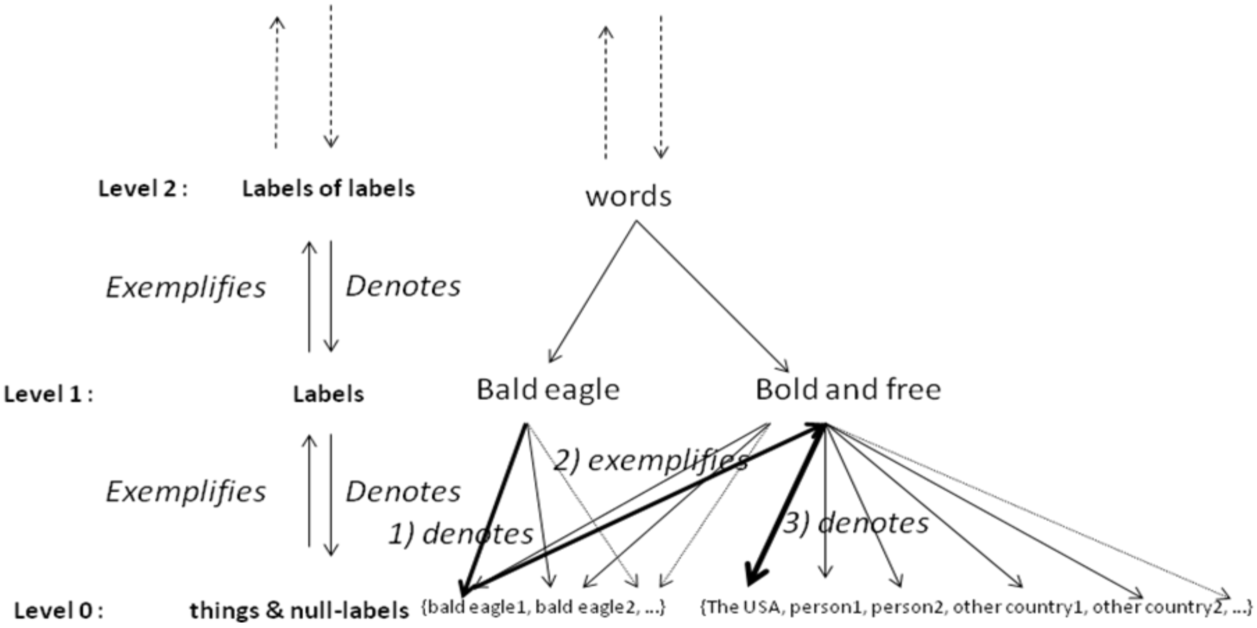
adapted from Goodman (1981) and Phan & Varenne (2010)

In Figure 4, one can see that we often use *labels of labels* such as the word "words". It serves to add another level upward and, in this specific case, to denote distributively each of the words we ordinarily use in the level below. There could be an infinite number of levels up because, through the reiterate adding of quotation marks, it is always possible to form a label of label of label and so on.

In this context, an example of a particular *chain of reference* is given by Goodman:

Figure 5 – Chain of reference to symbolize the USA

A picture of a bald eagle denotes a bird that may exemplify a label such as "bold and free" that in turn denotes and is exemplified by a given country (Goodman 1981:127).



As one can see in Figure 5, a chain can have 3 links. In our case, those are 3 successive and atomic[21] *routes of reference* which form the *chain*: 1. denotation, 2. exemplification, 3. denotation. What is striking here is the lateral move within the same level that is enabled through this remoteness of reference and this indirect way of referring. The move is from a given bald eagle (because, for instance, the bald eagle at the level 1 is a photograph of the given bald eagle1 occurring at the level 0) to a given country, the USA.

Computer simulations and denotational hierarchies

Computer simulations: a characterization

The characterization of CS we will give is not properly a definition as we will not show that no other practice can match this characterization too. But, this characterization seen as a necessary condition will be sufficient for our purpose.

In its broadest meaning, a computer simulation (or computational simulation[22]) can be characterized as a *strategy of symbolization within which there is at least 1) one step-by-step operation on symbols and 2) for which this operation involves at least one change of level of symbols.*

Moreover, such a strategy of symbolization has to present two major phases:

1st phase (*operative phase*): a phase of step-by-step operations running on symbols denoting entities

[21] A single link is an *atomic* route of reference in the sense that it is not made up of other routes of reference.

[22] As opposed to analog simulation, for which no step by step operation on symbols is necessary, *de jure*, a computational simulation can be perfomed with papers and pencils, without a computer. But, *de facto*, their spreading in science is due to the use of computers.

which are supposed to denote either real or fictional entities, reified rules or micro-behaviors in global phenomena.

2nd phase (*observational phase*): a phase of observation or measure or any mathematical or computational re-use of the results of the 1st phase (e.g.: the simulated "data" taken as input data for a model of pattern recognition or for another simulation, etc.). These results are taken as new symbols standing at another symbolic level than the one occupied by the symbols operating in the 1st phase.

If there is no change of levels of symbols at all, one is facing a formal procedure on discrete symbols not a simulation. Current softwares enabling formal calculus use the notational aspect (*i.e.* character indifference, syntactic unambiguity) of high-level programming languages. Thanks to these languages, a computer can be user-friendly and *emulate* some formal motor of deductions.

But, a CS is a strategy of symbolizing, *i.e.* of referring. As such, it may, but it does not have to, refer to something existing externally to the systems of symbols at stake, be it real or fictive. CS often use null labels. Hence, a CS may have a target system or not. This is the reason why there are two familiar and apparently incompatible meanings of CS in the literature: 1. a computation of a model (where a *model* is defined as a *dynamic formal construct possessing unity, formal homogeneity and simplicity*[23]), 2. a simulation in the sense of an imitation of a target system (Ören 2005 ; Yilmaz *et al.* 2006). In the former familiar characterization, CS has not to simulate any external target system. But, it still is a CS in that operates step-by-step and through a change of level of symbols.

A crucial aspect of a computer simulation is this emphasis on a *change of levels* for its operating symbols. In their conceptual presentation of a general theory of systems, (Klir & Elias 1985) particularly focus on the hierarchical aspect of the relationships between systems specifications[24]. Following this hierarchical presentation, (Zeigler *et al.* 2000) suggest seeing a *simulator* itself as a system specification standing on a level below the level of the initial model and, thanks to this change of level, generating the *behavior* of this model. In Zeigler *et al.* (2000), the change of level is crucial as a *simulator* is defined as "any computation system […] capable of executing a model to generate its behavior".

Hence, we see that even for an overall and theoretical standpoint as Zeigler's, executing a model by its micro-behaviors necessitates that we *change the level* of variables and of their correlated symbols specifications. Here, we find a notable agreement between our analysis of the hierarchy of systems of symbols in denotational hierarchies and this theoretical definition of computer simulations proposed by practitioners. From this agreement, it follows that a CS seems to be a question of levels of symbols and not only a matter of levels of languages. Nevertheless, such a characterization of a CS by Zeigler *et al.* still is grounded on the notion of model. From this, it logically follows that any CS can be reduced to a "calculus of a model" again. But, as we have already put it into perspective, this reduction is due to the prior adoption of a theoretical view on systems in general, of a general theory of systems, for which 1) models are seen as systems *i.e.* as formal constructs closed under composition, and 2) where only one DH (denotational hierarchy) appears and that incorporates the target system itself (in the case there is one).

> Due to the unique hierarchization and to the integration of all the target objects within the same hierarchy, a change of level can be seen as an explicitation of what is already there, but implicit. As a consequence, simulation cannot appear as anything other than a *simulation of model* as we defined it above (a *set of target objects* being always seen as a *system-model*). As underlined by Zeigler *et al.* (2000), "in the M&S context, one major form of systems analysis is computer simulation *which generates data under the instructions provided by a model*" (my emphasis) (Varenne 2010a:70).

In this closed systemic view, there is no external referring function for symbols of a CS. It entails that individual symbols of a CS do not refer to some things or aspects of things existing outside the specifications of the model and the simulator: there has to exist no other concurrent DH than

[23] This "simplicity" of a model is not absolute, but relative to our instrument of control, representation, conception and/or manipulation of the models which are available or preferred at a given time in the history of science. Today, it is often supposed to be the length or the readability of an equation (for a non-aided human mind). But it also can be a writeability in a preferred axiomatic, or a manipulability, a communicability, a publishability, a translatability or other rhetorical, instrumental and contextual reasons.

[24] For a brief recall, see Varenne (2010a:68-70).

the only one supposed to operate in the model and in its simulator. It is not surprising: a general system theory becomes mathematically fruitful to the extent that it entails such an internalization of external denotation functions of symbols[25]. Historically, this theoretical view on CSs by Zeigler and his colleagues can be understood. It came from practices that first appeared in operational research and in the specific *engineering sciences*, not in *empirical sciences* in general.

But, the problem is the following: In this system-theoretic perspective on CS, the paradigm that serves to think simulation implicitly remains *emulation* although *numerical simulations*, themselves, do not *emulate* the formal calculus of the model they simulate. They are based not on emulation, but on functions of *approximate* referring: The validity of a numerical CS of a model is based on available mathematical theorems that show that *approximate* atomic computations on micro-behavior of the model generated on discrete elements (or discrete differences) converge for this precise model. Hence, it is based on an accredited knowledge which is not indicated in the symbols themselves and which is external to the hierarchy. Moreover, it is this external accredited (mathematical) knowledge which gives its legitimacy to the functions of internal labeling in the DH which are used for the step-by-step computations. More precisely, this external knowledge legitimates that the names of variables distributively denote numerical tokens and that we have the right to see them, at the same time, as instances of such variables in the hierarchy. Consequently, if it is already the case for numerical CS, computer simulation, in general, cannot be a matter of *emulation*.

Furthermore, if we follow the strict characterization of CSs given by Zeigler *et al.*, it is not possible to explain further the difference between the strategy of symbolization of a numerical simulation of a mathematical model and the strategy of an agent-based simulation. Of course, at the level of the numerical electronic components or even at the level of the machine language or of micro-programming, all programs performed on a classical computer are *de jure* reducible to recursive statements in first-order logic. From this technical fact, it is sometimes concluded that all CSs are of the same kind. But, such an extreme viewpoint logically leads to some excessive and disputable claims such as: "all which is performed on computers is computer simulation" or "all formal calculi emulated by a computer are also computer simulations".

Our aim here is to explain why the vision that practitioners have on CSs is much richer and not logic oriented nor mono-leveled. And it is to show too that, for this aim, it is not necessary to make use of inner representations, of intensions, of cultural preferences or of deference. It is to show that an extensionalist approach of symbols suffices if we want to discriminate 1) between any running of a computer and a CS, and 2) between types of CSs.

By doing this, we follow Epstein (2006) in his answer to the classical objection made by *theoretical computer scientists*:

> In any event, the issue is not whether equivalent equations exist, but which representation (equations or programs) is most illuminating. To all but the most adept practitioners and perhaps to them as well, the recursive function representation would be utterly unrecognizable as a model of social interaction, while the equivalent agent model is immediately intelligible as such (Epstein 2006:xiv).

Hence, a prior problem is that many reflections on CSs often overlook two facts, whch are 1) the fact that not one, but *many* denotational hierarchies (DHs) are indeed operating in the same CS at the same time and 2) the fact that external DHs cannot be always and *a priori* reduced to internal DHs (because simulation is *not* emulation except for some computational simulations simulating computational systems) even if it is more simple and intellectually comfortable, from both a theoretical and epistemological viewpoint, to authoritatively internalize the functions of external referring.

Denotational hierarchies and types of computer simulations

In the following sections, we will show why the notion of *external denotational hierarchy* is relevant when we want to explain the epistemic diversity of current computer simulations. But, firstly, let us think more about our characterization of CS. Let us specify it further in front of empirical observations of existing practices of computer simulation.

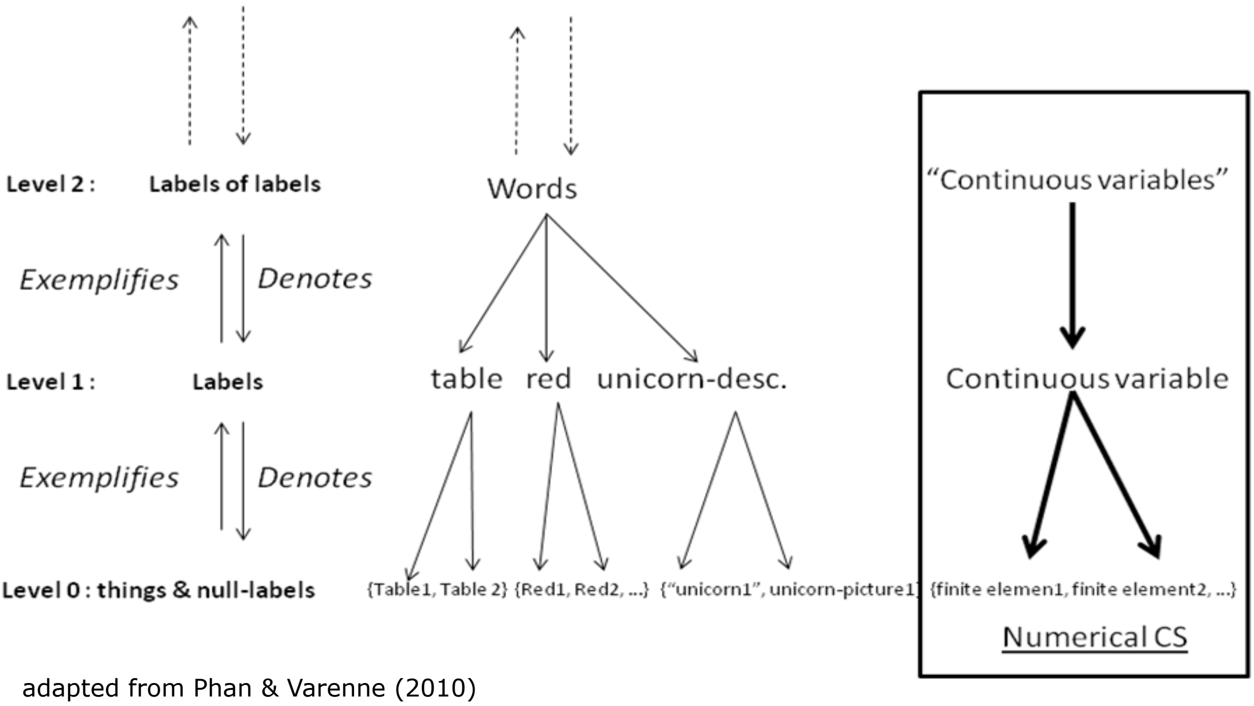
[25] See Varenne (2010a) for further discussions on this point.

In previous historical and empirical works[26], it has been suggested that we can find at least three different kinds of CS in the current practices of sciences:

1. A CS is *model-driven* or *numerical* when it proceeds from a unique internal DH. In such a CS, a unique model lies at the *Label level* of the DH. The change of level of such a CS follows a 2-links+computation *chain of reference* (see Figures 6 & 7). Namely: 1) a discretization of the continuous variables of the model which goes from labels of variables of the model to their exemplifications by numerical tokens in the level below, 2) a computation consisting in interactions between these exemplifying elements during the operative phase (at the same level), 3) a return to another level up which can be seen as the new labeling and denotating level for the aggregated results of the operative phase (observational phase using visualizing, measuring procedures,

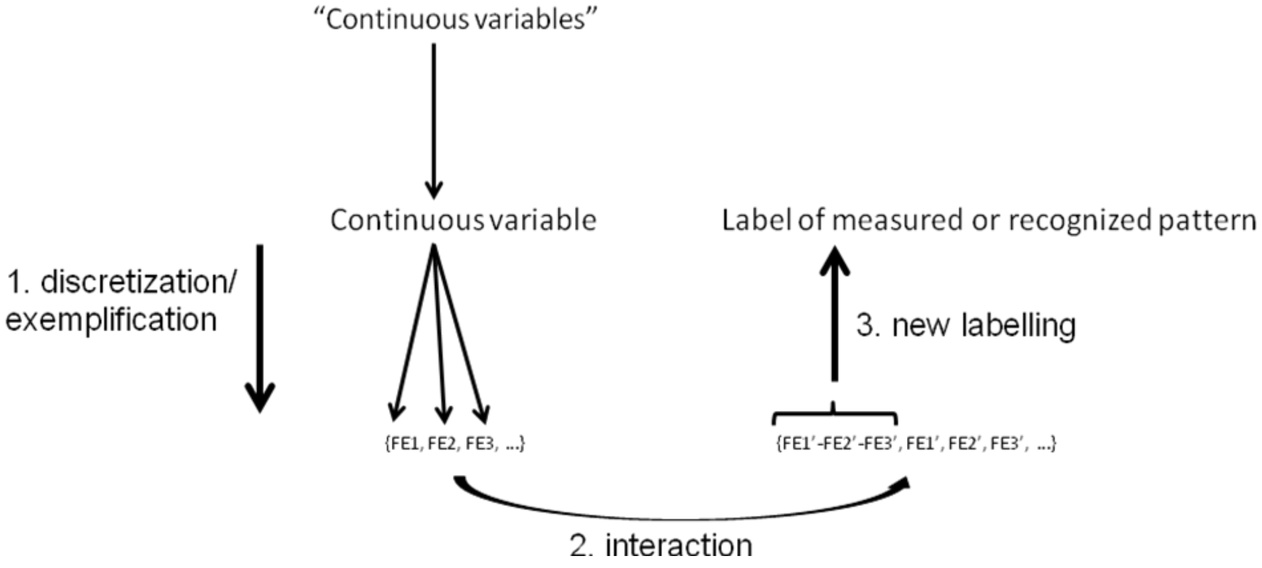
[26] Varenne (2003, 2007, 2009, 2010a), Phan & Varenne (2010).

Figure 6 – A Numerical Computer Simulation and its internal denotational hierarchy



adapted from Phan & Varenne (2010)

Figure 7- Chain of reference in a Numerical Computer Simulation

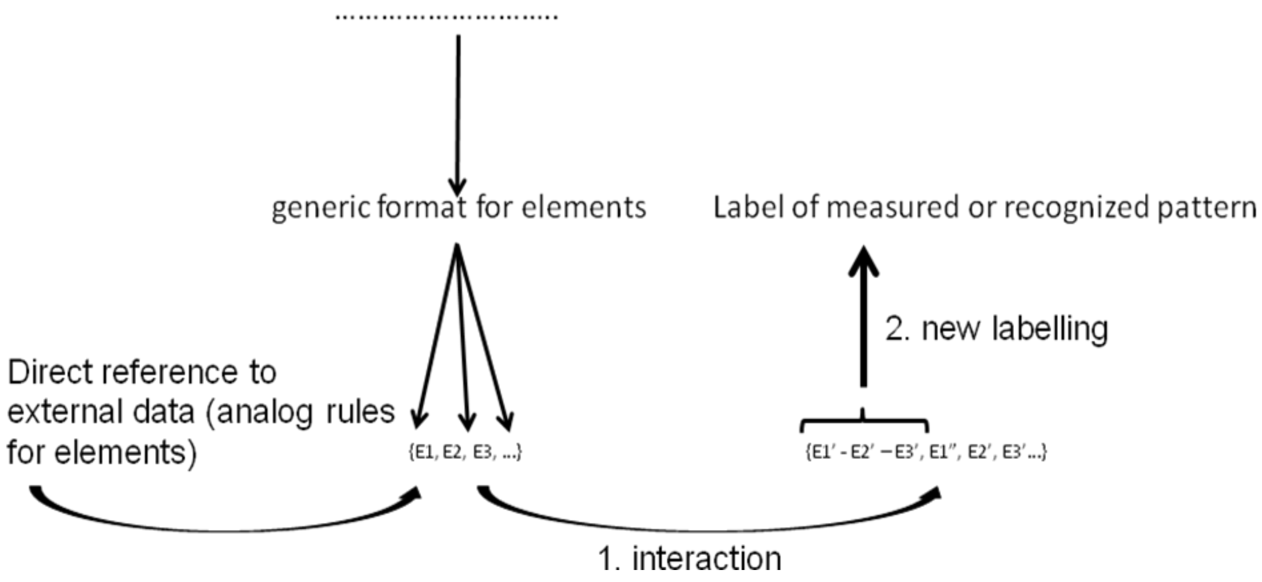

"Continuous variables"
Continuous variable
Label of measured or recognized pattern
1. discretization/ exemplification
3. new labelling
{FE1, FE2, FE3, ...}
{FE1'-FE2'-FE3', FE1', FE2', FE3', ...}
2. interaction

statistics, *etc*.). In the literature on CS, it is often noticed that what we call here "exemplifying elements" in such a model-driven or numerical simulation (be they finite elements or finite differences) can be unrealistic, such as are fictive discrete elements in computational fluid dynamics. They are said to be syntactic conventions which enable some approximate evaluations of the behavior of the model. This view is correct, precisely to the extent that *finite elements* of a numerical CS are *null-labels* in the internal DH of the CS. As such, they are *used in the computation* just the same way "unicorn" or "centaur" are used in ordinary language discourses and in their associated denotational hierarchies. Their function of denoting is denied. They stand at the bottom level of the DH. But their functional capacity of interaction as tokens is what is put at the forefront and used by the computational system.

In Figure 7, "FE" is an abbreviation of Finite Element. "FE1'-FE2'-FE3'" denotes the final aggregate that is formed as a result of the interactions between FE1, FE2 and FE3. In this aggregate, FE are noted FE' as step #2 takes time and leads to another moment of time. At step #3, such an aggregate is recognized or categorized, hence *labeled* (either by the computer program itself or by a human external observer or by an external device of signal or pattern recognition), so as to be newly denoted by the level above.

2. A CS is *rule-driven* or *algorithmic* when it does not start from an overarching unique aggregative model. It starts from symbols that are taken as null-labels in the internal DH. Those symbols exemplify some features of individual symbolic entities occurring in the internal DH. And they also and directly *refer to* properties of entities that are supposed to exist *outside* the internal DH of the computation. For instance, CSs using generative grammars, such as Chomsky's grammars for computational linguistics, rewriting systems or L-systems by Lindenmayer (1968a, 1968b) for computational developmental biology, are based on the assumption that elementary interaction rules between symbols directly refer to *analog rules* occurring between external entities such as phonemes or cells. Such a CS presents a 1-link+computation *chain of reference* (see Figure 8). The important point is that, contrary to model-driven CSs, the legitimacy of its exemplifying elements does not come from a discretization of pre-given labels belonging to the internal denotational hierarchy. But it comes from an external knowledge that each time 1) gives a verifiable meaning to the term "analog" in the expression *analog rules* and 2) that accordingly serves to legitimate the link between those elements that stand at the bottom of the *internal DH* and some field data and/or observations or measures that belong to some or to only one *external DH*. This *external hierarchy* is crucial as it works and operates in ordinary language too. As such, it serves to *root*

Figure 8- Chain of Reference in a Rule-Based Computer Simulation


....
generic format for elements
Label of measured or recognized pattern
2. new labelling
Direct reference to external data (analog rules for elements)
{E1, E2, E3, ...}
{E1' - E2' −E3', E1'', E2', E3'...}
1. interaction

the references[27] of its own labels through public procedures and through shareable instruments which give access to data. As recently underlined by Peschard (forthcoming), this access to data *via* instruments is an indirect one. But it is based on direct *causal interactions* between things, between properties of instruments and properties of real entities. On the contrary, the function of referring does not have to be based on such causal interactions: this is the reason why CSs are no direct substitutes for real experiments (*ibid.*) even if, at their level, they are *experiments* on the possible outcomes of the multifarious interactions between computational elements[28]. Nevertheless, thanks to this final function of *laterally referring* to an external DH, it remains possible for a *chain of reference of a CS* too to be a public, shareable and rigid one: no idiosyncratic intensions nor deference nor social enrollment process need to be convoked here to legitimate the knowledge that can be acquired when operating with a CS. Another important point on *rule-based CS* is that it operates on only one *internal DH*, because only one axiomatic is at stake: Hence, there are frequent confusions between such CS and some recursive mathematical models. This uniqueness of axiomatic is mostly seeable in generative grammars (Lindenmayer 1968a&b), in many elements-based computational approaches or even in some of the first rule-based knowledge systems (Davis & Lenat 1982).

3. A CS is *object-driven* or *software-based* when it starts from a deliberate objectification[29] of denoted entities (even if the target system is not objectual in that it can be conceptual or fictitious) and from a discretization of associated properties and behaviors. Such CS is facilitated by the use of high-level languages and of *softwares* dedicated to object-oriented modeling such as UML (Booch *et al.* 1998). Beware that this objectification does not necessarily entail any ontological commitments toward objects, concrete entities, as opposed to processes, for instance. From the viewpoint of the question of the epistemic status of CSs, its main goal is methodological. Such a multidimensional discretization of entities, properties and behaviors is mostly desirable as it enables the *co-computation* of a multiplicity of *internal denotational hierarchies*. Through individual-based CSs (in ecology) or agent-based CSs (in social sciences), these diverse *internal denotational hierarchies* can work and interact quasi-simultaneously[30] in the computation. What is common with rule-based CSs is that such CSs often start from exemplifying elements rather than from labels. But such CSs are sometimes *complex* as they are mixing not only denotational hierarchies, but also numerical simulations and rule-based simulations. For instance, even in a simple theoretical agent-based social simulation such as Sugarscape (Epstein & Axtell, 1996), the heterogeneity of a geographical environment (its content in sugar) can firstly be described through some continuous mathematical functions. Then, these functions are discretized and instantiated in discrete elements so that these elements of environment finally can interact with particular agents. For this procedure, we have a 2-links

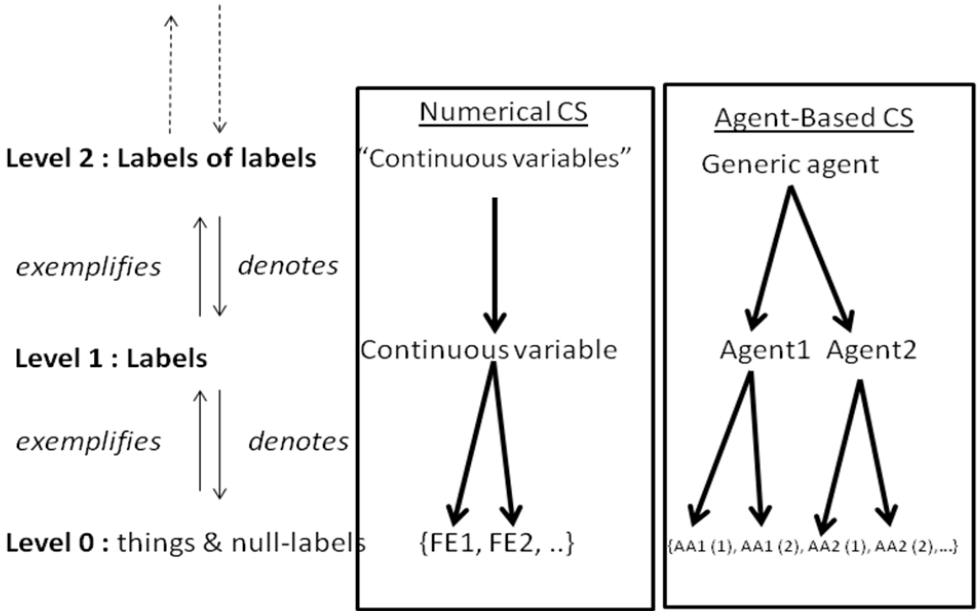
[27] As a nominalist, but relativist, Goodman (1976, 1978, 1981) had contested the idea that we could *root* the final references of our terms or symbols in an ultimate bed of real entities. This is a reason why he chose to entitle his famous article on chains of references "Routes of reference". Through this title, he clearly alluded to the book of Quine (1974), *The roots of reference*, while at the same time refusing Quine's view on this very point. Meanwhile, Scheffler (1997) has shown that adopting an extensionalist approach does *not* entail adopting relativism: we still can make a difference between the world and a world version. Followingly, our suggestion here, is to rejoin Quine on this possibility of sometimes *rooting* our symbols, while assuming at the same time that the *routes of reference* are multiple, complex and sometimes entangled.

[28] Note that, it is the very fact that computational elements are operating as *tokens* and not as *types* which implicitly troubles experiments-minded epistemologists. Indeed, this fact wrongfully encourages them to conceive computer simulations as having the same epistemic status as experiments made on those entities referred to by the symbols occurring in the CS. CSs are experiments in themselves. But they are no exact (neither emulating) substitute of real experimentations even if they symbolize them, sometimes even to some very precise and multifarious extent.

[29] The term *object* does not necessarily denote what Goodman (1977), for instance, calls *concreta*, but only what he calls *complexes*. "A concretum is a fully concrete entity in that it has among its qualities at least one member of every category within some sense realm" (ibid.:145). In this context of a constructionist philosophy of entities through *qualia* given by senses, a *complex* is an entity that has more that only one *qualia*. Analogously, and grounding our own argument of what is called an object in object-oriented programming, we can say that an *object* in the implementation of the internal denotational hierarchies of an object-driven CS is analogous to a *complex*: in order to be the result of an objectification in this sense, it does not have to refer to all the facets of its corresponding entities that are assumed in some real or fictional world but only to some of them.

[30] In classic computers (sequential machines), simultaneity of processes can only be simulated, *i.e.* not emulated, but approximately reproduced. On this important point, see Coquillard & Hill (1997).

Figure 9 – Numerical CS and Agent-Based CS with their denotational hierarchies



adapted from Phan & Varenne (2010)

chain of reference. But, these agents themselves are given features which have been directly identified thanks to their external direct reference to some experimental data or to some previous theoretical model of social mechanisms. For this procedure, we have a 1-link *chain of reference*.

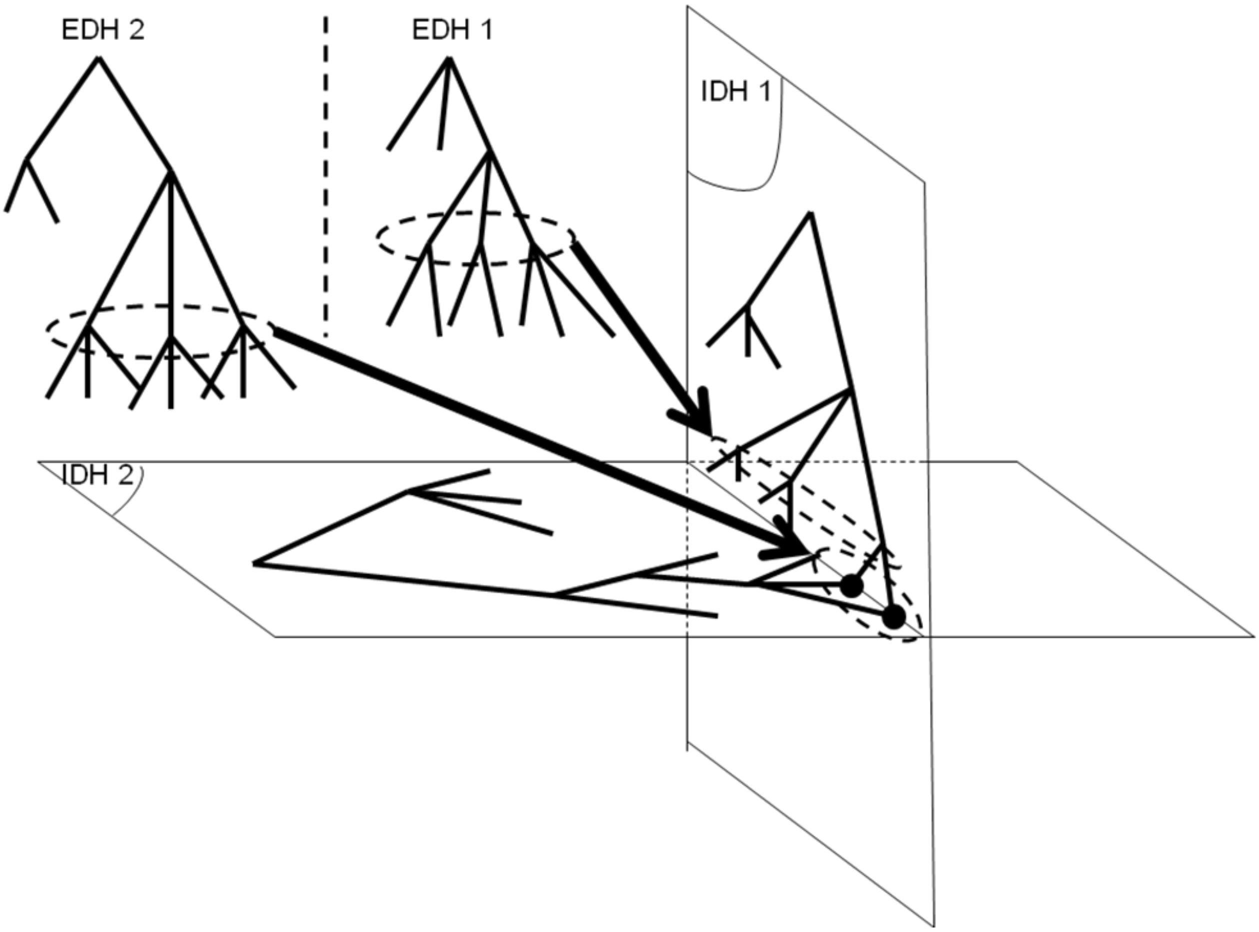
In Figure 9, "AA" is an abbreviation of "Aspect of agent" and "AA1 (1)" is an abbreviation of "Aspect #1 of agent #1".

Note that what we call "direct reference to external data" in Figure 8 refers to data that belong, themselves, to some denotational hierarchies which are assumed to be external to the hierarchy used by the computation itself. Its content can be numerical data, a qualitative pattern or some theoretical and/or a hypothetical representation of a physical, biological or social mechanism. As we soon will see, although it can be contested at first glance, it is fruitful to assume that these external data belong to some *denotational hierarchies* too. Furthermore, each level of a given internal DH of a CS can refer to symbols belonging to some level of another but *external DH*. The various ways of referring to an external DH for symbols in a CS is what we can characterize as the various procedures of *reference rootings*. This rooting has not to be taken in any absolute sense. Contrary to what Quine (1974) assumes in his own inquiry, it suffices to suppose here that such a rooting operates though a recognized and stabilized routine of extraction of data or of theorizing or of representing hypothetical mechanisms[31].

As we can see in Figure 10, tokens of interacting agent aspects are at the crossings of multiple Internal DH (IDH). Each of these multiple IDHs independently finds its rooting in some External DH (EDH). But these rootings do not necessarily occur at the same level. Rootings can occur at the bottom of the IDH or at another level. For instance, Figure 10 (IDH1's rooting in EDH1) shows that the rooting and qualitative or numerical identification of one aspect of agents has been made a level above the level of the interacting computational elements, whereas another IDH (IDH 2) can have its rooting at the elementary or "things or null-labels" level. Moreover, it is not necessary that a given level (*i.e.* with a given rank in the hierarchy) of an IDH refers to the same level (*i.e.* with the same rank) in the corresponding EDH. For instance, the bottom level of an IDH (see IDH2 in Figure 10) does not have to refer to any *qualia* level nor to any foundational level. It means that an EDH does not have to go down to some real things or *qualia* level. Even if they are said to be external, external denotational hierarchies do not have to be ontologically grounded: their hierarchical recognizable arrangement suffices to ground the agreement between

[31] As in computational *analytical sociology* for instance, see Demeulenaere (2011) on this point.

Figure 10 – Internal Denotational Hierarchies and their cross-references to External Denotational Hierarchies in an Object-Driven Computer Simulation



designers and practitioners of the CS, even if it is floating, ungrounded. It follows that an applied epistemology of CSs does not need to assume any constructionist ontology, even if it can make a fruitful use of its conceptual instruments, particularly on the complex functioning of the relationships between symbols and between things and symbols.

External reference in computer simulations as mention-selection

One could object that the relationship between EDHs and IDHs still is not clear. This is true. Until now, we have seen that all cases of *standing for* or of *symbolization* have to be thought *via* the assumption of a single pre-given (socially accepted or verifiable) hierarchy of denotations and exemplifications. But what about the relationships between symbols and/or entities belonging to *different* hierarchies? Is this relationship still a way of referring? When two denotational hierarchies are referring to each other, does this mean that they both belong to a third one? This solution would lead to useless complications.

Let us see if one finds some better clarification if one conceives external reference in computer simulations as a *mention-selection* relationship. As we had written in Section1.3, a symbol does not only denote (when it is *used*) its extension, but when it is not used, it still "*mentions* some other symbols which *stands at the same symbol level* and which are known to sometimes denote the same things". It is not necessary to assume that both the symbol and the other symbols it mentions-selects stand at the same level of the *same* denotational hierarchy. On the contrary, even if Scheffler (1997) does not enter into many more details on this question of the variety of extensions in relation with denotational hierarchies, it is coherent to assume that mention-selection can - and most frequently *has to* - occur between *different* denotational hierarchies. Hence, the split appearing in

Figure 2 can be interpreted as representing two different hierarchies. Accordingly, the *chain of reference* goes from one hierarchy to another and not only from one level to another in the same hierarchy.

Beware that it does not imply that any symbol can mention-select any other symbol in another DH, because an important constraint gives ground to this relationship: the symbol and what it mention-selects must have been taught simultaneously to be linked in a durable, shareable and public relationship of *mention-selection*. As has been shown by Scheffler (1997), mention-selection is a weaker relationship than iconicity even if they both entail some similar epistemic constraints, among which the constraint of simultaneous teaching. As we understand now, iconicity in its strongest sense[32] is very often, but wrongly confused with mention-selection.

Hence, we suggest that the relationship of external referring between internal DHs of a CS and external DHs are relationships of mention-selection. A sign of the fruitfulness of such a suggestion can be seen in the fact that practitioners are speaking of "agents" when they are speaking either of the tokens of formal constructs occurring in the running of the CS or of real "human agents" occurring within human society. Recall that, similarly, the word "man" both *denotes* men and *mentions-selects* representations of men in a picture. What is more, Scheffler (1979) and Scheffler (1997) insisted on the important fact that, contrary to classical symbolic ambiguity or linguistic puns, the *two chains of reference* occurring in a relationship of mention-selection have to occur simultaneously to be really operative. In the example of "a rose is a rose", we see that the sentence is meaningful if and only if both chains of reference are *simultaneously* available. Through that point, another sign of fruitfulness of our suggestion appears: To be grounded and justified, a CS has to use symbols that belong to some IDH and that denote or exemplify other symbols within this IDH and which corresponding mention-selected symbols *simultaneously* denote some verifiable extension *via* some public, shareable and reproducible rooting procedures.

Epistemological applications: a sample

In this section, we apply this extensionalist presentation of symbols, denotational hierarchies and chains of references in computer simulations of some classical and still disputed questions.

Chains of reference in multimodel-based, multiscale and integrative simulations

The role of axiomatic heterogeneity within CS

Thanks to our extensionalist approach, it is possible to explain further recent trends in the history of CSs. As shown in previous works[33], even if it can be interpreted from an ontological point of view, in particular within research programs fascinated by some atomistic ontology (for instance, in *some* programs belonging to AI, ALife or computational biology), the strategy of increasing the number of links in the chains of references in a CS first of all has a *technical and methodological motivation*.

The important point is that the function of referring *via exemplifying* enables all the different IDHs to meet on the same elementary symbolic level. Through that, each IDH reaches a common operative level: mathematically heterogeneous formal representations of things, of agents, of aspect of things are objectified and discretized and treated step-by-step so as to become *co-computable*. After these co-computations, each IDH can go upward on its own again so that it refers more intelligibly or more "illuminatively" (Epstein 2006:xiv) to a given aspect of physical, biological or social reality. From that, it is possible to understand why the *multi-model based* approach in *integrative computer simulations of plant growth* succeeded comparatively to the current collapse of classical mathematical modeling in this domain[34].

Since the 1990s, thanks to discretization and exemplification *via* object-driven CSs, a more exact modeling of *composite things* and of phenomena possessing a *strong inner heterogeneity* has become possible. This success can be seen in the recent spreading of CSs of growth and morphogenesis processes particularly occurring in biological, environmental and social systems[35].

[32] Iconicity traditionally involves at least one identity or feature between the denoted and the denoter. But see Phan & Varenne (2010) on some recent and weaker meaning of "iconicity" and on its application to DHs in CSs.

[33] Varenne (2007, 2008, 2009b).

[34] On this particular point, see Varenne (2007).

[35] See, for instance: Michaewicz (1997), Kohler & Gumerman (2000), Kohler & van der Leeuw (2007).

Scales and hierarchies

Today, more and more efforts are done towards so-called *multi-scale computer simulations*. It is particularly the case in the domains of brain simulation, human body simulation, ecological simulation, social simulation and bio-geo-chemical-simulation. A confusion can occur from the meeting between our epistemological use of the term *hierarchy* and this technical trend. It could seem as if we were speaking about the hierarchies which are supposed to really exist in nature such as ecological scales. But our presentation makes us understand that these denotational hierarchies do *not* have to be taken as ontologically grounded. It suffices to assume that they are, to some verifiable extent, *rooted* in available data through verifiable procedures. The term "scale" denotes a kind of real arrangements and ontological dependence between levels of entities. Contrary to the term "level", when we use the term "scale" we assume that the corresponding level possesses a naturality which is not necessarily present in denotational hierarchies[36]. As they are goal-oriented, but shareable hierarchical frameworks of things *and* symbols, not only of things, neither IDHs nor EDHs do have to rely upon any realistic assumption concerning the leveling of things.

Integrative simulations and objectification

As we have already noticed, what we call *objectification* is not a complete reification. That is: when we say that an object-driven CS relies on some objectification, it does not mean that its IDH have to completely symbolize an entity *concretely* existing in some assumed reality. As a consequence, it becomes understandable to epistemologists that, when practitioners say they are designing *integrative CS*, they are not saying that they are designing any total nor exhaustive CS of a given portion of reality. More modestly and technically, they are saying that they are doing no numerical CS but, instead, an object-driven CS which itself relies on *many cross-ways of referring* to diverse types of data and/or scales of data.

Cross-validations in computer simulations

An important technical problem of complex CSs, particularly of agent-based ones, is their validation[37]. The notion of validation is rather ambiguous in the literature. Roughly speaking and most frequently, a model is said to be validated when it represents or reproduces correctly the elements, the mechanisms and/or the behaviors of some target system. Many ambiguities here come from the fact that we seek either *representations* of things (or mechanisms) or *reproductions* of behaviors. We will not enter into this debate here.

It is note worthy that if a research program chooses to use computer simulation and not statistical modeling, it certainly means that its applied epistemology of models does not rely on some classical instrumentalism such as that of Friedman (1953). A CS enables some detailed representations of hypothetical mechanisms at some micro-scale as well as representations of (in the sense of *reference to*) intermediary data. Regarding this advantage, and its counterbalance in the drawback of its difficult manipulation, a CS is rarely used as a pure phenomenological model of prediction. As the motivation for modeling through a CS is generally more ambitious than it is for pure phenomenological modeling, practitioners are often facing this classical objection coming from arguments on the under-determination of models by data and on the dissymmetry between deduction and induction: "even if you have increased the realism of the simulation by compelling it to follow some detailed and intermediary data, you can never show that there exist no other plausible mechanisms for the same phenomenon". The modelers Ormerod & Rosewell (2009) present the dispute between economists on this point in the following terms:

> Both the present authors, and the experience is widely shared among ABM [Agent-Based Models] modelers, have encountered from economists a view which can be summarized as follows: you have presented one set of behavioural rules to explain your chosen phenomenon, but there must be many such sets which do this, so how do you know yours are correct? Some economists even go on to imply that it is easy to construct successful ABMs, an opinion which merely reveals their ignorance of the difficulties involved. The fact that they do

[36] Some mathematically minded modelers keep on assuming that mathematical hierarchies (such as those built in the mathematical theory of category for instance) are of natural kind too. As a consequence, the epistemology of theses modelers often overlooks the distinction between denotational hierarchies in models and simulations, on the one hand, and ontological scales, on the other hand.

[37] See Hill (1995), Amblard *et al.* (2007), Ormerod & Rosewell (2009).

> not appear to appreciate that the rules of behavior incorporated into much standard economic analysis are of very special kind which does not stand up well to experiment should not blind us to the need to provide a sound basis for choice of the rules of behavior for any individual ABM application (Ormerod & Rosewell 2009:135)

But, if it is surely desirable, what could be a "sound basis" for such a choice? This is where, to some extent, the authors' view meets our own on the *chain of reference* and on some rootings of the denotational hierarchies in data. Because they answer:

> One key test is that the behavioural rules should be capable of justification using evidence form outside the model. The better this evidence, the more credible the rules (*ibid.*, our emphasis).

Notice that the evidences coming from "outside the model" are not supplementary logical arguments. They are *evidences* in the plain sense of the term, as are data. The authors here try to coin some extensionalist approach for which symbols of the CS could directly denote some external evidence. From their viewpoint, then, two possibilities appear: either evidences are present, or they are absent. If they are absent, one has to choose *simple rules* so that they can be at least easily understood. If they are present, the CS must be designed to meet with the highest accuracy these evidences. But no additional concepts nor analysis on this meeting are given by the authors.

The authors go on with a more polemic tone:

> The inherent methodology of ABMs is far more scientific than that of conventional economics. We identify a set of empirical macro-features, and plausible behavioural micro rules are designed from which the macro properties emerge. This is much more scientific methodology than econometrics, for example, much of which is mere curve fitting and is not modeling in any real sense of the word (*ibid*: 136).

We will not enter here in the other and difficult question of the comparison between econometrics and ABM in economics. By quoting these lines, we underline the importance recognized by practitioners of these *multiple ways of referring* to direct evidences for complex computer simulations. Many other practitioners, such as Axtell (2000), Edmonds (2007), Moss (2001, 2005) or Conte (2000), more or less agree on this idea that when evidences are available, a CS 1) has to be more detailed and 2) has to refer to these evidences so as to increase its credibility.

But, although increasingly more modelers publish these kinds of arguments (alluding to the reference of symbols in IDHs to intermediary evidences in EDHs), not many economists nor sociologists, for instance, are convinced: it seems to them as if another kind of fitting, but at a micro-level this time, had replaced the curve-fitting of econometricians. So, their questions become the following: Where does the epistemic novelty stand? And, where should the increase in credibility come from in agent-based computer simulations?

A more structured and discriminating answer comes from what Moss and Edmonds (2005) have called *cross-validation* of agent-based models and simulations. From our viewpoint, this method is interesting in that, even if its concepts are technical rather than epistemological, it is the first one to explicitly rely on a *kind of multi-level way of referring* through CS. A sketch of this method and its correlated arguments is given by Neumann (2009):

> The notion of cross-validation implies that agent-based models, validated at the micro level, allow the generation of statistical patterns at the macro-level which are in accordance with empirical observations: there are a widespread number of cases where the aggregate data exhibits unpredictable cluster of volatility. In fact, this feature can be reproduced by the means of agent-based simulation models. Moss and Edmonds stress that "this result does not occur because we tune our models to produce this kind of time series but rather seems to be a consequences of the characteristics we put into our models" (Moss & Edmonds 2005:1121).
>
> […]
>
> Hence the notion of cross-validation implies the assumption that validations on the micro- and macro levels are independent of one another. This however, means that the statistical macro patterns cannot be derived analytically from the designs of the agents on the micro-level […] (Neumann 2009:81).

Hence, *cross-validation* designates a method which makes use of two independent qualitative and quantitative *ex post* comparisons (not *ex ante* fittings) between external data and some results (time series) of the simulation extracted at two different *levels of its denotational hierarchies*.

To this remarkable auto-justification and reinforcement of the justification of their own practice by practitioners, we suggest adding that it is not necessary to think of *cross-validation* as relying on only two levels. The vision of these two levels comes from a traditional approach in social sciences: the individual agents and the groups or institutions (states, churches, etc.). Note that, in this context, this restriction itself is reinforced by the persistent confusion (denounced above) between scales and levels. The implicit syllogism is the following: 1) If you assume that there are only two major and real scales in the social reality (individuals and groups or institutions); 2) If you assume that levels in IDH are directly referring to (denoting) real scales and not to intermediary or other levels of information or formulation of the social; 3) You should conclude that any validation of your social CS has to be worked upon at only two levels.

Of course, from a general point of view on validation in complex software-based CS, one can extrapolate that cross-validation does not have to limit itself to such a two-level approach except if our aim is to take part to some theoretical researches on emergence as it is the case in the Neumann's (2009) article. As there can be a multiplicity of IDHs in the same object-driven CS (see section 2.2), the number of possible cross-validations shall increase. What will remain a problem of course is the question of mutual independency of such validations. But as many formal representations and symbols in those different IDHs are axiomatically heterogeneous, it can be inferred that such independence won't be too rare nor too difficult to show.

On computer simulations and the multiplicity of their epistemic statuses

In this section, we will give only a sample of the consequences this extensionalist approach of the function of referring *via* computer simulations can have for an applied epistemology of CS. We will insist on the ability it gives us not to reject, but to understand and classify some current epistemological arguments on CS.

Two popular meanings of CS in the literature: simulation of model, model of simulation

As we have seen, an extensionalist approach gives a meaning to the notion of *chains of reference via* computer simulation. It appeared that a computer simulation always involves at least one change of level in some denotational hierarchy. Thanks to this characterization, we have had the possibility to reconcile the two popular, but divergent meanings of the expression "computer simulation" in the literature: 1) the step-by-step calculation of a mathematical model and 2) the simulation of a target system through a model of simulation. Thanks to our characterization, we can understand that the following sentence, which is often implicit in many articles of practitioners, is not completely absurd nor meaningless: "We simulated our model of simulation".

This sentence has a meaning in that its two replicas of the same term "simulation" do not refer to the same extension. With the first replica, by "simulation" we mean a *simulation of model*: the emphasis is put on the *exemplification* of elementary behaviors of a pre-given model, *i.e.* on the change of level within an IDH. With the second replica, by "simulation" we put the emphasis on the fact that a CS is also a strategy of *external referring* and that, as such, it entails some IDHs that are designed to refer to (*i.e.* to *mention-select*) some external denotation hierarchies.

Hence, if we apply our general characterization of a CS, we find precise grounds that justify the two familiar uses of the same term to denote two practices that seem to be so different in nature. And, it becomes explainable that we go on merging in the same terms these two ways of referring to different practices.

On some divergent standpoints on CSs: CSs as experimental or theoretical arguments?

From this overlooked ambiguity, there follows a number of divergent epistemological standpoints on CS. We will not propose here a systematic classification, but only a few examples thanks to which we will show the power of analysis that our extensionalist view can give.

First, if you prefer to overlook the referential function of CS and if you insist on the fact that a CS involves a *change of level* in a unique DH (this uniqueness being questionable as we have seen), you will argue that CSs are only computations of some pre-given models and, consequently, that CSs essentially are homogeneous to theoretical arguments (see *e.g.* Stöckler 2000).

Second, if one prefers to insist on the fact that this change of level is felt by the practitioners to be an obliged path when they want to get some results of some complicated computation, one will argue that CSs are a *kind of experiment*. From this viewpoint, a CS is a kind of experiment in the sense that its results are *surprising us* (Morgan 2005) as we do not have the possibility to go faster (*i.e.* with less mathematical nor intellectual steps) *via* some abridged formal calculus and deduction: surely, as simulation is no emulation, the necessary internal *chain of reference* cautiously has to be followed by a computer-aided series of operations on symbols.

Third, if one substantiates first the notion of "thought experiment" and if one prefers to draw a strong parallel between thought experiments and machine delegated experiments on discretized elements standing at the bottom of the IDH (*things or null-labels level*), one will insist on another experimental aspect of CS (Di Paolo et al. 2000).

Fourth, if one focuses on the fact that the specific epistemic power of CSs, contrary to mathematical modeling, does not mainly rely on any global nor *formal validation*, but on a certain amount of term-to-term *referential relationships* between features of physical objects and features of the simulation model elements (*i.e.* on the ability of a CS to *mention select symbols or things of an EDH through symbols of an IDH*), then one will choose to overlook the classical argument that says that, in a CS, the model is an insurmountable obstacle between symbols and reality. And, implicitly grounding one's argument on the *rootings of references* in CSs, one will prefer to claim that some CSs present a kind of "materiality" not "in the machine itself, but rather in the simulation model" (Morrison 2009:45). From that, one will conclude "that the object of inquiry is also the physical system and that the knowledge we obtain is of that system" *(ibid.)*.

We do not have enough room to give more examples of such conceptual elucidations on seemingly contradictory theses on CS. Many other epistemological claims on the status of CSs can be classified and render complementary rather than contradictory thanks to our referentialist and extensionalist viewpoints. The important point here is that such a view and its following elucidations are tolerant to many epistemological arguments. This does not mean that "anything epistemological goes"; we hope not, having shown that they do not involve any accommodating pragmatism grounded on some conceptual vagueness. Through its analytical tools, this view tries to explain further and for each case the very reason why a given epistemological thesis is justified, and why it can prevail from other avowed viewpoints on CS.

Inscriptionalism and the experimental nature of CSs

For the last application of this view on the question of the status of CS, it is necessary to go a little further than simple extensionalism and to adopt a strict inscriptionalist approach. *Inscriptionalism* is an extreme form of extensionalism that "takes for granted only the individual tokens and the individual things that may be denoted" (Scheffler 1979:9). For instance, Goodman still gives a role to types of symbols, whereas Scheffler tries to avoid types.

We have previously insisted on the fact that symbols at stake in a CS are not *types* of symbols but rather *tokens of symbols*. They are events of symbols, such as replicas of word-types in a text. CSs are machine delegated computations on symbols. As a machine does not work with classes nor ideas nor intensions, it is important to understand that a *real computer* operates at every moment of its operating time, not on types of symbols, but on singular *inscriptions of symbols*. These inscriptions exist in the causal world as they are sets of real electrical levels which are real properties of some electronic components called gates (N.AND or N.OR gates). Hence, there is a possibility to argue - as did Fetzer (2004) - that symbols in a CS plainly are submitted to causality. As a consequence, one can infer that a CS is a real experiment in itself, because it is an experiment in a causal world. In this context, an inscriptionalist view permits to make a clear distinction between a symbol type and its tokens: it shows where exactly the "good" reasons that support this thesis lie.

In general, the objections to this view rely on the claim that these electronic levels are not *uncontrolled variables* contrary to what we implicitly assume when we say that they operate the same way real properties operate in a real experiment. Another answer consists in recalling that a CS is a *strategy of symbolization*. It consists in focusing on the symbolic counterparts of electrical levels. And it is true too that, when a computer functions correctly, these levels are exactly *emulated* by a symbolic system written in first-order logic. This symbolic system is completely equivalent to electrical levels but at another level in that its way of operating use types not token. Consequently, from our viewpoint, and if we assume that any computer works in perfect conformity to its theoretical representation, it can be accepted too that the ontological separation between a word-type and its material correlate has no epistemic consequence.

Notwithstanding, our view enables to persist in thinking that a token remains different from its type and that a token *may not* be reliably nor constantly linked to its type, just the way an equivalent logical system has to assume it is when applied to computers. Thanks to this persistent and understandable distinction, for instance, the problem of the correctness and of the equivalence of effective compilers operating in different machines[38] still can be understood. It can be thought as an open empirical question and, then, in all its real, *i.e.* technical, dimensions without being submitted to any previous reduction to logic only (Varenne 2009:47).

Conclusion

This paper has presented an extensionalist analysis of computer simulations and a sample of its epistemological applications. As this approach is still not used nor well represented in the literature on CSs, the first sections have given an overview of its consequences for the philosophy of symbols in general. Then, an argument in favor of a large characterization of computer simulations (CSs) has been proposed. Whereas, most characterizations of CSs focus on their being a calculation of a model, where the model is assumed to be written in an unique mathematical language, our characterization suggests focusing on a more fine-grained level, namely on the *symbols* that operate in any CS. Of course, these symbols operate and interact each time at a given level. As such, computers not only simulate, but they also can *emulate* some formal calculi. But the first conceptual refinement that our characterization permits is the necessary introduction of *denotational hierarchies* of symbols, null-labels and things in any CS.

Accordingly, a CS can be distinguished from a formal calculus 1) as it always involves at least one *change of level* in its denotational hierarchy and 2) as its functioning as a plain CS entails that it really works as a *strategy of symbolization*. This second condition implies that its denotational hierarchy constantly and diversely has to *refer to* things or symbols that are *different from* and *external to* (at least in principle) the symbols operating in the implementation.

Another fruitful consequence of our analysis relies on the fact that this necessity *to refer to* something does not imply that practitioners, nor stakeholders nor epistemologists of CS shall possess and share such questionable entities as *inner representations* or *intensions*. It is possible to make these analyses without any recourse to something other than *extensions*.

Through the analytical instrument of *denotational hierarchy*, it has been possible too to explain precisely the nature and the diversity of the *chains of reference* that can occur in different kinds of CSs. Thanks to this, it has been possible to give conceptual grounds to a first classification of CSs.

Finally, this paper has proposed some applications of this analytical view to some classical questions in epistemology of CSs. The first important outcome is that this approach can explain on what conceptual reasons the methodology of *cross-validation* can be grounded. Hence, it gives us clues on why this approach probably will succeed and spread in the coming years, particularly in the growing domain of integrative and multi-scale CSs. The second important outcome consists in the fact that this extensionalist view is tolerant to many epistemological theses on CS. Finally, it appears fruitful too in that it permits to uncover *what aspects* of CS authors have implicitly chosen to take into account when they try to show that, for instance, CSs are not experiments or are only a kind of experiment or even a kind of theoretical argument.

[38] Leading itself to the debate on the nature of the verification of programs: empirical or theoretical? See Varenne (2009a) and its bibliography on this debate.

References

Amblard, F, Bommel P. & Rouchier J. (2007) Assessment and Validation of multi-agents Models, *in* Phan & Amblars (2007), pp.93-114.

Axelrod R. (1997/2006) Advancing the Art of Simulation in the Social Sciences. In: Conte, R., Hegselmann, R., Terna, P. (eds.), *Simulating Social Phenomena*, Berlin, Springer-Verlag, pp.21-40. Updated version in Rennard, J.P. (ed.), *Handbook of Research on Nature Inspired Computing for Economy and Management*, Hersey, PA: Idea Group, 2006.

Axtell R. L. (2000) Why agents? On the varied motivations for agent computing in the social sciences, *in* Macal, M. & Sallach, D. (eds.), *Proceedings of the workshop on agent simulation: Applications, models, and tools*, Chicago, IL: Argonne National Laboratory, pp.3-24.

Bedau M. A. (1998) Philosophical content and method of Artificial Life*, in* Bynum & Moor, J.H. (eds), *The Digital Phoenix: How computers are changing philosophy*, Oxford, Basil Blackwell, pp. 135-152.

Colburn T. (2004), Methodology of Computer Science, *in* L. Floridi (ed.), *The Blackwell Guide to the Philosophy of Computing and Information*, New-York, Blackwell, 2004, pp. 318-326.

Colburn T. & Shute G. (2007), Abstraction in Computer Science, *Minds and Machines* 17(2), pp. 169–184.

Conte R. (2000), The necessity of intelligent agents in social simulation , *in* Ballot G. & Weisbuch, G. (eds.); *Application of Simulation to Social Sciences,* London, Hermes, pp. 19-38.

Booch G., Rumbaugh J. & Jacobson I. (1998), *The Unified Modeling Language User Guide*, New York, Addison Wesley.

Copeland B.J. (2004), Computation, *The Blackwell Guide to the Philosophy of Computing and Information*, L. Floridi (ed.), New York, Blackwell Publishing, 2004, pp. 3-17.

Coquillard, P. and Hill D.R.C. (1997), *Modélisations et simulations d'écosystèmes*, Paris, Masson.

David N., Caldas H. and Coelho, H. (2010) (eds), *Epistemological Perspectives on Simulation – III*, special section in the *Journal of Artificial Societies and Social Simulation*, 13 (1), 14, http://jasss.soc.surrey.ac.uk/13/1/14.html.

David N., Sichman J.S. and Coelho H. (2005) The Logic of the Method of Agent-Based Simulation in the Social Sciences: Empirical and Intentional Adequacy of Computer Programs, *Journal of Artificial Societies and Social Simulation* 8(4)2 http://jasss.soc.surrey.ac.uk/8/4/2.html.

Davis R. & Lenat D.B. (1982), *Knowledge-Based Systems in Artificial Intelligence*, New York, Mc Graw Hill.

Demeulenaere P. (ed.), *Analytical Sociology and Social Mechanisms*, Cambridge, Cambridge University Press.

Di Paolo E.A. *et al.* (2000) Simulation Models as Opaque Thought Experiments, *Artificial Life VII*, Proc. of the 7th Intern. Conf. on Artificial Life, ed. by Mark A. Bedau *et al.*, Cambridge, MIT Press, pp. 497-506.

Dietrich M. R. (1996) Monte-Carlo Experiments and the Defense of Diffusion Models in Molecular Population Genetics, *Biology and Philosophy*, vol. 11, n°3, July 1996, pp. 339-356.

Edmonds B. (2007). « Simplicity is Not Truth-Indicative ». In Gershenson, C.et al. (2007), *Philosophy and Complexity*. World Scientific, 65-80.

Edmonds B. & Moss S. (2001). « The Importance of Representing Cognitive Processes in Multi-Agent Models », in: Dorffner, G., Bischof, H. and Hornik, K. (eds.), *Lecture Notes in Computer Science*, 2130:759-766.

Edmonds B. & Moss S. (2005), « From KISS to KIDS – an 'anti-simplistic' modelling approach ». In P. Davidsson et al. (Eds.): Multi Agent Based Simulation 2004. Springer, *Lecture Notes in Artificial Intelligence*, 3415:130–144.

Elgin, C.Z. (1983), *With Reference to Reference*, Indianapolis, Hackett Publishing.

Epstein J.M. (2006), *Generative Social Science*, Princeton, Princeton University Press.

Epstein J.M. & Axtell R. (1996), *Growing artificial Societies*, Cambridge, The MIT Press.

Ferber J. (1999), *Multi-agent Systems: an Introduction to Distributed Artificial Intelligence,* Reading, MA: Addison-Wesley Publishing Company.

Ferber J. (2007), Multi-agent Concepts and Methodologies, *in* Phan, D. & Amblard, F. (2007), pp.7-34.

Fetzer J.H. (2000), “Philosophy and Computer Science: Reflections on the Program Verification Debate, *in* Bynum & Moor, J.H. (eds), *The Digital Phoenix: How Computers are Changing Philosophy*, Oxford, Blackwell Publishers, pp. 253-273.

Forrester J. W. (1968) *Principles of Systems*, Cambridge, MIT Press.

Keller E. F. (2002a), *Making Sense of Life. Explaining Biological Development with Models, Metaphors and Machines*, Cambridge Mass., Harvard University Press, 2002.

Keller E. F. (2002b), Models, Simulation and ‘Computer Experiments’, *The Philosophy of Scientific Experimentation*, ed. by Hans Radder, University of Pittsburgh Press, 2002, pp. 198-215.

Friedman M. (1953) *Essays on Positive Economics*, Chicago, Il., University of Chicago Press.

Galison P. (1987) *How Experiments End*, Chicago, Il., University of Chicago Press.

Galison P. (1996), Computer Simulations and the Trading Zone, *in* Galison P. & & Stump D.J. (eds.), *The Disunity of Science*, Stanford University Press, pp. 118-157.

Galison P. (1997), *Image and Logic*, Chicago, Il., University of Chicago Press.

Gilbert N. & Troitzsch K. Eds. (1999) *Simulation for the Social Scientist*, Philadelphia, Open University Press.

Gilbert N. (2007), Computational Social Science – Agent-based Social simulation, *in* Phan & Amblard (2007), pp. 115-133.

Goodman N. (1976), *Languages of Art: An Approach to a Theory of Symbols*, Indianapolis, Bobbs-Merrill, 1st edition: 1968.

Goodman N. (1977), *The Structure of Appearance*, 3rd edition (1st edition : Harvard UP, 1951), Boston, Reidel.

Goodman N. (1978), *Ways of Worldmaking*, Indianapolis, Hackett Publisher.

Goodman N. (1981), Routes of reference, *Critical Inquiry*, 8(1), pp. 121-132.

Guala F. (2002), Models, Simulations, and Experiments, *in* Magnani, L. & Nersessian, N. J. (eds.), *Model-Based Reasoning: Science, Technology, Values*, NY, Kluwer, pp.59-74.

Guala F. (2003), Experimental Localism and External Validity, *Philosophy of Science*, 70, pp.1195-1205.

Hailperin M., Kaiser B. & Knight K. (1999), *Concrete Abstraction – An Introduction to Computer Science using Scheme*, Scarborough (Canada), Thomson Publishing.

Hacking I. (1983), *Representing and Intervening*, Cambridge Ma, Cambridge University Press.

Hartmann S. (1996), The world as a process, *in* Hegselmann, R., Müller, U. & Troitzsch, K. (eds.) *Modelling and simulation in the social sciences from the philosophy of science point of view,* Dordrecht, Kluwer, pp.77-100.

Hartmann S. (1995), Simulation, *Enzyklopädie Philosophie und Wissenschaftstheorie*, vol. 3, Verlag Metzler, Stuttgart, pp. 807-809.

Hill D.R.C. (1995), Verification and Validation of Ecosystem Simulation Models, *Summer Simulation Conference*, July 24-26, Ottawa, Canada, pp. 176-182.

Hill D.R.C. (1996), *Object Oriented Analysis and Simulation*, Addison-Wesley.

Humphreys P. (1990), Computer Simulations, *PSA* (Philosophy of Science Association) 1990, vol. 2, pp. 497-506.

Humphreys P. (2004), *Extending Ourselves: Computational Science, Empiricism, and Scientific Method*, Oxford, Oxford University Press.

Klir G. & Elias D. (1985), *Architecture of Systems Problems Solving*, New York, Plenum Press.

Kohler T.A. & Gumermann G.J (2000), *Dynamics in Human and Primate Societies*, Oxford, Oxford University Press.

Kohler T.A. & van der Leeuw S.E. (2007), *The Model-Based Archaeology of Socionatural Systems*, Santa Fe, School for Advanced Research Press.

Küppers, G., Lenhard, J. & Shinn, T. (2006), Computer Simulation: Practice, Epistemology, and Social Dynamics, *in* J. Lenhard, G. Küppers & T. Shinn (eds.), *Simulations: Pragmatic Construction of Reality*, Springer, pp. 3-22.

Lindenmayer A. (1968a), Mathematical models for cellular interactions in development. I. Filaments with one-sided inputs, *Journal of Theoretical Biology*, 1968, 18, p. 280-299.

Lindenmayer A. (1968b), Mathematical models for cellular interactions in development. II. Simple and branching filaments with two-sided inputs, *Journal of Theoretical Biology*, 1968, 18, p. 300-315.

Livet P. (2007), Towards an Epistemology of Multi-agent Simulations in social Sciences, *in* Phan & Amblard (2007), pp.169-193.

Mäki U. (2005), Models are experiments, experiments are models, *Journal of Economic Methodology*, 12(2), pp. 303-315.

Michalewicz M. (1997), *Plants to ecosystems. Advances in Computational Life Sciences*, Collingwood (Australia), CSIRO Publishing, 1997.

Morgan M.S. (2005), Experiments versus Models: New Phenomena, Inference and Surprise, *Journal of Economic Methodology*, 12, pp. 317-329.

Morgan M.C. & Morrison M. (eds.) (1999), *Models as Mediators*, Cambridge Ma., Cambridge University Press.

Morrison M.C. (1998), Experiment, *in* Craig, E. (ed.) *The Routledge Encyclopaedia of Philosophy*, London, Routledge, pp. 514-518.

Morrison M.C. (2009), Models, measurement and computer simulation: the changing face of experimentation, *Philosophical Studies,* 143, pp. 33-57.

Moss S. & Edmonds B. (2005), Sociology and Simulation: Statistical and Qualitative Cross-Validation, *American Journal of Sociology*, 110(4), pp. 1095-1131.

Neumann M. (2009), Emergence as an Explanatory Principle in Artificial Societies, *in* Squazzoni (ed.), pp. 69-88.

Neumann (von) J. (1961), *The general and logical Theory of Automata (1948)*, in A. H. Taub *Collected Works of von Neumann*, Volume V, Oxford, Pergamon Press, 1961, pp. 288-328.

Ören T.I. (2005), Toward the Body of Knowledge of Modeling and Simulation. *in Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*, Orlando, Florida; paper 2025, 2005, pp. 1-19.

Ormerod P. & Rosewell B. (2009), Validation and Verification of Agent-Based Models in the Social Sciences, *in* Squazzoni (ed.), 2009, pp. 130-140.

Peck S.L. (2004) Simulation as experiment: a philosophical reassessment for biological modeling, *Trends in Ecology & Evolution*, 19(10), pp. 530-534.

Peschard I. (forthcoming), Is Simulation an Epistemic Substitute for Experimentation?, Proc. of *the International Colloquium on Computer Simulations*, held at Marseille, FRUMAM-CEPERC, France, June 2010, forthcoming.

Phan D. and Amblard F. (eds.) (2007), *Agent Based Modelling and Simulations in the Human and Social Sciences*, Oxford, The Bardwell Press.

Phan D. & Varenne F. (2010), Agent-Based Models and Simulations in Economics and Social Sciences: from conceptual exploration to distinct ways of experimenting, *Journal of the Artificial Societies and Social Simulation (JASSS)*, 13(1), 2010, http://jasss.soc.surrey.ac.uk/13/1/5.html

Quine W. v. O. (1974), *The Roots of Reference*, La Salle, Open Court Publishing.

Quine W.v.O. (1981), *Mathematical logic*, Harvard University Press, 1st edition: 1940.

Reiss J. (2011) A Plea for (Good) Simulation: Nudging Economics Toward an Experimental Science, *Simulation & Gaming*, 2011, 42(2), pp. 243-264.

Rohrlich F. (1990), Computer Simulation in the Physical Sciences, *PSA* (Philosophy of Science Association) 1990, vol. 2, pp. 507-518.

Scheffler I. (1979), *Beyond the Letter*, London, Routledge & Kegan.

Scheffler I. (1997), *Symbolic Worlds*, Cambridge (UK), Cambridge University Press.

Schruben L.W. (1980), Establishing the credibility of simulations, *Simulation*, March 1980, pp. 101-105.

Squazzoni F. (ed.), 2009, *Epistemological Aspects of Computer Simulation in the Social Sciences,* Berlin, Springer Verlag, 2009.

Stöckler M. (2000), On Modeling and Simulations as Instruments for the Study of Complex Systems, *Science at Century's End*, Proc. of the Pittsburgh/Konstanz Colloquium in the Philosophy of Science, October 1997, Univ. of Pittsburgh Press, pp. 355-373.

Tesfatsion L. (2002), Agent-Based Computational Economics: Growing Economies from the Bottom Up, *Artificial Life*, 8(1), March, pp.55-82.

Varenne F. (2001), What does a computer simulation prove? *in* Giambiasi, N. & Frydman, C. (eds.) *Simulation in industry - ESS 2001*, Proc. of the 13th European Simulation Symposium, SCS Europe Bvba, Ghent, pp. 549-554.

Varenne F. (2003), La simulation conçue comme expérience concrète, *in* Müller J.P. (éd.), *Le statut épistémologique de la simulation*, Paris , éditions de l'ENST, 2003, pp. 299-313.

Varenne F. (2007), *Du modèle à la simulation informatique,* Paris, Vrin.

Varenne F. (2008), Modèles et simulations : pluriformaliser, simuler, remathématiser. In: Kupiec, J.J., Lecointre, G., Silberstein, M., Varenne, F. (eds.) *Modèles Simulations Systèmes*. Paris, Syllepse, pp.153-180 ; reprinted in F. Varenne & M. Silberstein (eds), *Modéliser & simuler. Épistémologies et pratiques de la modélisation et de la simulation*, Paris, Éditions Matériologiques, 2013, pp. 299-328.

Varenne F. (2009a), *Qu'est-ce que l'informatique ?*, Paris, Vrin.

Varenne F. (2009b), Models and Simulations in the Historical Emergence of the Science of Complexity. *in* Aziz-Alaoui, M.A. & Bertelle, C. (eds), *From System Complexity to Emergent Properties*, Berlin, Springer, pp. 3-21.

Varenne F. (2010a), Framework for Models & Simulations with Agents in regard to Agent Simulations in Social Sciences: Emulation and Simulation, *in* A. Muzy, D. Hill & B. Zeigler (eds), *Modeling & Simulation of Evolutionary Agents in Virtual Worlds*, Clermont-Ferrand, Presses Universitaires Blaise Pascal, pp. 53-84.

Varenne F. (2010b), Les simulations computationnelles dans les sciences sociales, *Nouvelles Perspectives en Sciences Sociales*, 5 (2), 2010, pp. 17-49.

Winsberg E. (1999), Sanctioning models: the epistemology of simulation, *Science in Context*, 12, pp. 275-292.

Winsberg E. (2009), A Tale of Two Methods, *Synthese*, 169 (3), pp. 575-592.

Yilmaz L., Oren T. & Aghaee N. G. (2006), Intelligent agents, simulation, and gaming. *Simulation & Gaming*, Vol. 37, No. 3, September 2006, pp. 339-349.

Zeigler B. P., Praehofer H. and Kim Tag G. (2000), *Theory of Modeling and Simulation. Integrating Discrete Event and Continuous Complex Dynamic Systems*, New York, Academic Press, 2000, 1st edition: 1976.

Zeigler B. P., Muzy A. and Yilmaz L. (2009), Artificial Intelligence in Modeling and Simulation, *in* Meyers R. A. (editor-in-chief), *Encyclopedia of Complexity and System Science*, Heidelberg, Springer, 2009, pp. 344-368.

Working Papers : la liste

Hervé Le Bras, Jean-Luc Racine & Michel Wieviorka, *National Debates on Race Statistics: towards an International Comparison*, FMSH-WP-2012-01, février 2012.

Manuel Castells, *Ni dieu ni maître : les réseaux*, FMSH-WP-2012-02, février 2012.

François Jullien, *L'écart et l'entre. Ou comment penser l'altérité*, FMSH-WP-2012-03, février 2012.

Itamar Rabinovich, *The Web of Relationship*, FMSH-WP-2012-04, février 2012.

Bruno Maggi, *Interpréter l'agir : un défi théorique*, FMSH-WP-2012-05, février 2012.

Pierre Salama, *Chine – Brésil : industrialisation et « désindustrialisation précoce »*, FMSH-WP-2012-06, mars 2012.

Guilhem Fabre & Stéphane Grumbach, *The World upside down,China's R&D and innovation strategy,* FMSH-WP-2012-07, avril 2012.

Joy Y. Zhang, *The De-nationalization and Re-nationalization of the Life Sciences in China: A Cosmopolitan Practicality?*, FMSH-WP-2012-08, avril 2012.

John P. Sullivan, *From Drug Wars to Criminal Insurgency: Mexican Cartels, Criminal Enclaves and Criminal Insurgency in Mexico and Central America. Implications for Global Security,* FMSH-WP-2012-09, avril 2012.

Marc Fleurbaey, *Economics is not what you think: A defense of the economic approach to taxation,* FMSH-WP-2012-10, may 2012.

Marc Fleurbaey, *The Facets of Exploitation,* FMSH-WP-2012-11, may 2012.

Jacques Sapir, *Pour l'Euro, l'heure du bilan a sonné : Quinze leçons et six conclusions,* FMSH-WP-2012-12, juin 2012.

Rodolphe De Koninck & Jean-François Rousseau, *Pourquoi et jusqu'où la fuite en avant des agricultures sud-est asiatiques ?,* FMSH-WP-2012-13, juin 2012.

Jacques Sapir, *Inflation monétaire ou inflation structurelle ? Un modèle hétérodoxe bi-sectoriel,* FMSH-WP-2012-14, juin 2012.

Franson Manjali, *The 'Social' and the 'Cognitive' in Language. A Reading of Saussure, and Beyond,* FMSH-WP-2012-15, july 2012.

Michel Wieviorka, *Du concept de sujet à celui de subjectivation/dé-subjectivation*, FMSH-WP-2012-16, juillet 2012.

Nancy Fraser, *Feminism, Capitalism, and the Cunning of History: An Introduction,* FMSH-WP-2012-17 august 2012.

Nancy Fraser, *Can society be commodities all the way down? Polanyian reflections on capitalist crisis,* FMSH-WP-2012-18, august 2012.

Marc Fleurbaey & Stéphane Zuber, *Climate policies deserve a negative discount rate,* FMSH-WP-2012-19, september 2012.

Roger Waldinger, *La politique au-delà des frontières : la sociologie politique de l'émigration*, FMSH-WP-2012-20, septembre 2012.

Antonio De Lauri, *Inaccessible Normative Pluralism and Human Rights in Afghanistan,* FMSH-WP-2012-21, september 2012.

Dominique Méda, *Redéfinir le progrès à la lumière de la crise écologique,* FMSH-WP-2012-22, octobre 2012.

Ibrahima Thioub, *Stigmates et mémoires de l'esclavage en Afrique de l'Ouest : le sang et la couleur de peau comme lignes de fracture,* FMSH-WP-2012-23, octobre 2012.

Danièle Joly, *Race, ethnicity and religion: social actors and policies,* FMSH-WP-2012-24, novembre 2012.

Dominique Méda, *Redefining Progress in Light of the Ecological Crisis,* FMSH-WP-2012-25, décembre 2012.

Ulrich Beck & Daniel Levy, *Cosmopolitanized Nations: Reimagining Collectivity in World Risk Society,* FMSH-WP-2013-26, february 2013.

Xavier Richet, *L'internationalisation des firmes chinoises : croissance, motivations, stratégies,* FMSH-WP-2013-27, février 2013.

Alain Naze, *Le féminisme critique de Pasolini, avec un commentaire de Stefania Tarantino*, FMSH-WP-2013-28, février 2013.

Thalia Magioglou, *What is the role of "Culture" for conceptualization in Political Psychology? Presentation of a dialogical model of lay thinking in two cultural contexts*, FMSH-WP-2013-29, mars 2013.

Byasdeb Dasgupta, *Some Aspects of External Dimensions of Indian*

Economy in the Age of Globalisation, FMSH-WP-2013-30, april 2013.

Ulrich Beck, *Risk, class, crisis, hazards and cosmopolitan solidarity/risk community – conceptual and methodological clarifications*, FMSH-WP-2013-31, april 2013.

Immanuel Wallerstein, *Tout se transforme. Vraiment tout ?*, FMSH-WP-2013-32, mai 2013.

Christian Walter, *Les origines du modèle de marche au hasard en finance*, FMSH-WP-2013-33, juin 2013.

Byasdeb Dasgupta, *Financialization, Labour Market Flexibility, Global Crisis and New Imperialism – A Marxist Perspective*, FMSH-WP-2013-34, juin 2013.

Kiyomitsu Yui, *Climate Change in Visual Communication: From 'This is Not a Pipe' to 'This is Not Fukushima'*, FMSH-WP-2013-35, juin 2013.

Gilles Lhuilier, *Minerais de guerre. Une nouvelle théorie de la mondialisation du droit*, FMSH-WP-2013-36, juillet 2013.

David Tyfield, *The Coal Renaissance and Cosmopolitized Low-Carbon Societies*, FMSH-WP-2013-37, juillet 2013.

Lotte Pelckmans, *Moving Memories of Slavery: how hierarchies travel among West African Migrants in Urban Contexts (Bamako, Paris)*, FMSH-WP-2013-38, juillet 2013.

Amy Dahan, *Historic Overview of Climate Framing*, FMSH-WP-2013-39, août 2013.

Rosa Rius Gatell & Stefania Tarantino, *Philosophie et genre: Réflexions et questions sur la production philosophique féminine en Europe du Sud au XX*^{e} *siècle (Espagne, Italie)*, FMSH-WP-2013-40, août 2013.

Angela Axworthy *The ontological status of geometrical objects in the commentary on the Elements of Euclid of Jacques Peletier du Mans (1517-1582)*, FMSH-WP-2013-41, août 2013.

Pierre Salama, *Les économies émergentes, le plongeon ?*, FMSH-WP-2013-42, août 2013.

Alexis Nuselovici (Nouss), *L'exil comme expérience*, FMSH-WP-2013-43, septembre 2013.

Alexis Nuselovici (Nouss), *Exiliance : condition et conscience*, FMSH-WP-2013-44, septembre 2013.

Alexis Nuselovici (Nouss), *Exil et post-exil*, FMSH-WP-2013-45, septembre 2013.

Alexandra Galitzine-Loumpet, *Pour une typologie des objets de l'exil*, FMSH-WP-2013-46, septembre 2013.

Hosham Dawod, *Les réactions irakiennes à la crise syrienne*, FMSH-WP-2013-47, septembre 2013.

Gianluca Manzo, *Understanding the Marriage Effect: Changes in Criminal Offending Around the Time of Marriage*, FMSH-WP-2013-48, GeWoP-1, octobre 2013.

Torkild Hovde Lyngstad & Torbjørn Skarðhamar, *Understanding the Marriage Effect: Changes in Criminal Offending Around the Time of Marriage*, FMSH-WP-2013-49, GeWoP-2, octobre 2013.

Gunn Elisabeth Birkelund & Yannick Lemel, *Lifestyles and Social Stratification: An Explorative Study of France and Norway*, FMSH-WP-2013-48, GeWoP-3, octobre 2013.

Franck Varenne, *Chains of Reference in Computer Simulations*, FMSH-WP-2013-51, GeWoP-4, octobre 2013.

Olivier Galland & Yannick Lemel, avec la collaboration d'Alexandra Frenod, *Comment expliquer la perception des inégalités en France ?*, FMSH-WP-2013-52, GeWoP-5, octobre 2013.

Position Papers : la liste

Jean-François Sabouret, *Mars 2012 : Un an après Fukushima, le Japon entre catastrophes et résilience*, FMSH-PP-2012-01, mars 2012.

Ajay K. Mehra, *Public Security and the Indian State*, FMSH-PP-2012-02, mars 2012.

Timm Beichelt, *La nouvelle politique européenne de l'Allemagne : L'émergence de modèles de légitimité en concurrence ?*, FMSH-PP-2012-03, mars 2012.

Antonio Sérgio Alfredo Guimarães, *Race, colour, and skin colour in Brazil*, FMSH-PP-2012-04, july 2012.

Mitchell Cohen, *Verdi, Wagner, and Politics in Opera. Bicentennial Ruminations*, FMSH-PP-2012-05, may 2013.

Ingrid Brena, *Les soins médicaux portés aux patients âgés incapables de s'autogérer,* FMSH-PP-2013-06, avril 2013.

Thalia Magioglou, *Refaire l'Europe ou refaire le « monde » ? Un commentaire sur l'ouvrage : « Refaire l'Europe avec Jürgen Habermas »*, FMSH-PP-2013-07, septembre 2013.

Samadia Sadouni, *Cosmopolitisme et prédication islamique transfrontalière : le cas de Maulana Abdul Aleem Siddiqui,* FMSH-PP-2013-08, septembre 2013.

Alexis Nuselovici (Nouss), *Étudier l'exil,* FMSH-PP-2013-09, septembre 2013.