Emergence as Leverage and Non-Algorithmic Approaches in Agent-Oriented Software

Boldur E. Bărbat *, Sorin C. Negulescu **, Sorin Pleşca ***

* “Lucian Blaga” University of Sibiu, Faculty of Sciences, bbarbat@gmail.com

** “Lucian Blaga” University of Sibiu, Faculty of Engineering, sorin_negulescu@yahoo.com

*** “Politehnica” University of Timişoara, Faculty of Automation and Computers, plescasorin@yahoo.com

It is simplicity that is difficult to make.
BERTOLT BRECHT

Abstract: Asserting that current approaches to agent-oriented systems intended for open, heterogeneous, dynamic and uncertain environments are either unaffordable (with scarce resources) or ineffective (as regards user expectations), the paper aims at proposing a generic software framework based on *emergence* as main paradigm (within a blend with conventional ones, to attain a lever effect) and on *non-algorithmic approaches* to treat uncertainty at lower echelons (to suit complex application requirements even in very dynamic environments). This target is split into four specific objectives: a) investigating the relationships between complexity and emergence from the standpoint of modern artificial intelligence; b) showing that structural complexity can be dealt with through simulated emergence (e.g., via stigmergic coordination) and cognitive complexity through emulated emergence (e.g., via self-aware agents); c) investigating the (in)adequacy of logics and prediction methods used to handle uncertainty due to future contingents; d) outlining the path for developing affordable non-algorithmic mechanisms to deal with effectively. Since the emphasis is here on computer science aspects, design and implementation details about the mechanisms will be given in future papers. At this research stage, the results seem very promising and reveal significant potential for transdisciplinary projects (the main fields involved: complexity theory, synergetics, trivalent logics, cognetics).

Keywords: Emergence; Uncertainty; Open, heterogeneous, dynamic and uncertain environments (OHDUE); Agent-oriented software engineering (AOSE); Stigmergy; Agent self-awareness.

Boldur E. Bărbat M. Sc. in Electronic Engineering, postgraduate specialising in Programming, Ph.D. in Digital Computers. Currently: "Lucian Blaga" University Sibiu, Faculty of Sciences, full professor; "Politehnica" University Timişoara: advisor for doctoral studies in Computer Science. Disciplines: Artificial Intelligence, Software Engineering and related fields. Books: nine. Tutorials, textbooks: 11. Papers, articles: about 60 in the last five years (over 50 in English, almost all about agents). International Programme Committees: 17.

Sorin C. Negulescu has a M. Sc. degree in Computer Science and Automatic Control and a M. Sc. in Quality Management. Currently he is a doctoral student in Industrial Engineering. He is a teaching assistant and his research interests include ontologies, agent-oriented system engineering, anthropocentric interfaces and multi-agent systems.

Sorin Pleşca graduated at the "Lucian Blaga" University of Sibiu (Faculty of Science) 2007. He is now a doctoral student in Computer Science at the “Politehnica” University of Timişoara, Faculty of Automation and Computers. His research interests are stigmergic coordination, agent and multi-agent systems.

1. Introduction

“If "complexity" is currently the buzzword of choice for our newly minted millennium – as many theorists proclaim – "emergence" seems to be the explication of the hour for how complexity has evolved. Complexity, it is said, is an emergent phenomenon. Emergence is what "self-organizing" processes produce. Emergence is the reason why there are hurricanes, and ecosystems, and complex organisms like humankind, not to mention traffic congestion and rock concerts. Indeed, the term is positively awe-inspiring.” [14].

Moreover, as repeated recently in [8], IT environments, except for some trivial applications, are now *open* and *heterogeneous* (the resources involved are unalike and their availability is not warranted), *dynamic* (the pace of exogenous and endogenous changes is high) and *uncertain* (both information and its processing rules are revisable, fuzzy, uncertain and intrinsically non-deterministic – as every stimuli generator). Indeed, in the Internet era of “computing as interaction” [1], deterministic applications are practically vanishing.

On the other hand, the rules for human-agent interaction can and should be set by users (at least while we have the Demiurgic privilege of shaping agents as we like it!) [11]. However, most major agent-based systems are either too complex (to be affordable with scarce resources) or too ineffective (to meet even average problem requirements). As a result, end-user acceptance is low.

The paper aims at proposing a generic software framework based on *emergence* as main paradigm (within a blend with conventional ones, to attain a lever effect) and on *non-algorithmic approaches* to treat uncertainty at lower echelons (to suit complex application requirements even in very dynamic environments). The research of this middle-range undertaking is conducted from two slightly different perspectives – albeit both agent-oriented: the first concerns *concepts* and *approaches* for application development whereas the second focuses on (adapted or new) *software mechanisms*. Thus, the target is split into two pairs of objectives, each one comprising investigation (emphasis on computer science aspects) and taking advantage of its results (emphasis on agent-oriented software aspects); the four specific objectives are: a) investigating the relationship between complexity and emergence from the standpoint of modern artificial intelligence (AI); b) showing that structural complexity can be dealt with by simulated emergence (e.g., via stigmergic coordination) and cognitive complexity by emulated emergence (e.g., via self-aware agents); c) investigating the (in)adequacy of logics and prediction methods used to handle uncertainty due to future contingents; d) outlining the path for developing affordable non-algorithmic mechanisms to deal with. (Thus, it is metaphorically, in line with the book title: "Emergence: The Connected Lives of Ants, Brains, Cities and Software" [24].)

Related work, history, and approach as regards the branches of the undertaking are described recently (2007) in [6] [8] [9] [29]; to weaken redundancy, here they are skipped over. As a result, the rest of the paper is structured as follows: Section 2 expounds the *rationale* via *explaining the title*. Section 3 addresses the first pair of objectives examining *complexity as malady*, and proposing *emergence as antidote*. The next two sections concentrate on the second pair, showing the cumulative negative effect of *misunderstood concepts AND unaffordable or unsuited tools* (Section 4) and suggesting the direction for developing suitable *non-algorithmic mechanisms* giving two *examples* (Section 5). *Conclusions* (both factual and general) and directions of *future work* (Section 6) close the paper.

2. Rationale. Explaining the Title

For the sake of conciseness, the premises, criteria, context, and motives as well as their corollaries are not separated in conceptual categories but asserted clustered around the topic they address, i.e. the title key phrases. Thus, the rationale is "distributed" in the subsections.

2.1. Emergence. What kind of?

"What kind of?" has here a twin meaning: to clarify the connotations of this syncretic concept [20] [23], and to present the distinct paths followed by the undertaking to exploit emergence – i.e., simulating or emulating it.

The vagueness of the very concept and the renewed interest to capitalize on it are expressed in the title of [14]: "The re-emergence of "emergence": a venerable concept in search of a theory." The same work investigates the deep conceptual interference between emergence and complexity, showing that this re-emergence is due to "the growth of scientific interest in the phenomenon of complexity and the development of new, non-linear mathematical tools – particularly chaos theory and dynamical systems theory – which allowed scientists to model the interactions within complex, dynamic systems in new and insightful ways". Some definitions/connotations of "emergence" together with comments that are relevant for this paper are taken from [14]: "some of the confusion surrounding the term "emergence" might be reduced (if not dissolved) by limiting its scope. Rather than using it loosely as a synonym for synergy, or gestalt effects, or perceptions, etc., I would propose that emergent phenomena be defined as a "subset" of the vast (and still expanding) universe of cooperative interactions that produce synergistic effects of various kinds, both in nature and in human societies. In this definition, emergence would be confined to those synergistic wholes that are composed of things of "unlike kind" […] In other words, emergent effects would be associated specifically with contexts in which constituent parts with different properties are modified, re-shaped or transformed by their participation in the whole. In these terms, water and table salt are unambiguous examples of emergent phenomena. […] Can we explain consciousness as an emergent property of certain kinds of physical systems?"

In short, *emergence* is used in its both meanings mentioned above: a) the broader sense in the phrase "simulated emergence" (in the context of stigmergic control[1]); b) the "subset" sense in the phrase "emulated emergence" (in the context of agent self-awareness). The main reason to use both expressions was given in [8]: "Albeit both governed by emergence, the two kinds of undertaking are quite different:

---

[1] Synergy achieved through stigmergic control was called "stigsynergy" [13]. However, since a recent Google search showed that the term was not used by other authors (the only five mentions stem from [13] and [11]), the term will not be used in this paper (and maybe dropped in the future).

whereas when *simulating* common ant behaviour, emergence is expectable also from artificial ants and – despite obvious nonlinearity – their performances can be improved "incrementally" […] when *emulating* emergence the basic argument about "holism versus reductionism" transcends theory and becomes crucial for applications. Indeed, here are hidden not only the internal processes of self-organization (unanswered "Why?"), but also any "stigmas" (unanswered "How?")".

2.2. Why "as Leverage"?

The lever metaphor suggests here affordability. The problem was addressed in [8] asserting that, as regards advanced IT, it is generally accepted that academic research in East-European countries is still limited not by scientific potential but rather by financial or logistic boundaries; likewise, "synergy as leverage" was defended there, focusing on stigmergic control. However, two related aspects need further analysis: necessity and affordability of emergence-based approaches.

Necessity. For highly complex, dynamic, large-scale systems emergence is rather unavoidable since the interactions within such systems have emergent effects that must be mirrored in their IT images. Such systems being outside the scope of this paper, the question is: must be emergence considered when developing agent-oriented software for systems of average complexity? The answer is, almost certainly, *no* for both paths referred to: stigmergic control is replaced by "Cut and Try" procedures and agent self-awareness (without embodiment in robots) is a novelty. When generalising the question to cover all kind of emergent effects, the answer is still *no* because:

"The ability of current models fully to portray emergence in all its possibilities has been questioned […]. Reasons range from […] to the claim that the behaviour observed in real complex biological and social systems is uncomputable […]. Emergence by its nature is problematic to model. It is the product of interconnections and interaction making it dynamic and unpredictable; entities, interactions, their environment and time are key contributors to emergence, however there is no simple relationship between them. For example, how do novel system entities, such as the appearance of life, eyes or language appear?" [27].

Affordability[2]**.** Not just stigmergic control but all biologically inspired paradigms model massively parallel societies/systems (anthills, chromosomes, brains, etc.); thus, they are affordable (only) through simulation. On the contrary, assessing self-awareness through an "affordability filter" a question arises: is it appropriate to consider it as a relevant agent feature when many other strong agency characteristics are missing, even in current large-scale agent-based systems? Yes, because: system complexity makes it desirable [2] [3] [15] [26] and agent technology makes it possible [1] [17] [25] [34]. Thus, the software engineering challenge is to design affordable tools.

What should be levered? In mechanics, the tradeoff is *force* against *time*. Here it is the other way around: time is gained giving up the "force of perfection". In this context *perfection* means the dream of mathematicians – and algorithm-oriented software developer too – to deal with well-defined (if possible, monocriterial) problems, complete information, accurate data, acceptable time restrictions, low risk, conventional business, etc. and to give optimal solutions (at most Pareto optimality) through scores of exact data (if possible, output offline and sequential). In short, in real-world applications (mostly multicriterial, online, and distributed, supplied with incomplete, fuzzy, and/or uncertain information – arriving in parallel, in huge amounts and in unpredictable moments –, in the context of critical response time, high risk, virtual enterprises, etc.) the challenge is to *manage situations* (under bounded rationality [31] [32]), since there is no time to *solve* (accurately) *problems*. If a solution is needed, it must arrive "just in time" and be acceptable suboptimal; hence, emergence shall lever speed and simplicity (as antonym to complexity). That is why the title had to go on.

2.3. Non-Algorithmic Approaches

Albeit not "currently the buzzword of choice" as emergence already is, *non-algorithmic approach* is a core feature of the undertaking, and, hence, a cardinal issue of the rationale because almost all what happens in OHDUE and is important for applications cannot be modelled algorithmically, first because it is uncertain (as second reason is probably intrinsic complexity). That applies for emergence too: "Among

---

[2] In [8] affordability referred to university research where an undertaking is affordable if it proves workable as a project ending with a few experimental models validated, at most, "in vitro" (for instance, within the scope of a PhD thesis). Here the restrictions are more severe since the paper refers to software engineering.

other things, complexity theory gave mathematical legitimacy to the idea that processes involving the interactions among many parts may be at once deterministic yet for various reasons unpredictable […] Emergence does not have logical properties; it cannot be deduced (predicted)" [14]. Here, "non-algorithmic" suggests that [29] [11]: a) Conventional algorithms are not anymore *program backbone* (since in the era of "computing as interaction" deterministic applications are irrelevant – at least those affordable on usual configurations). b) Conventional algorithms are not anymore the main *programming instrument* (they are hidden in scripts or in procedures easily reached in a host of libraries). c) Higher order thinking is nonalgorithmic (the path of action is not fully specified in advance). d) Not only "algorithmic reasoning", but any algorithmic interaction of analog beings within analog environments seems unnatural (see the example in Section 5).

Non-algorithmic approaches in the history of this undertaking, regarding diverse aspects of *uncertainty* or *temporal dimension* in agent-oriented applications designed for OHDUE are [10]: a) *Implicit non-deterministic approach*: a1) *affective computing* (asynchronous reaction to environment stimuli, emotion as asymmetric temporal function, controlling ethical agent behaviour; a2) *user-driven heuristics* (even in stigmergic coordination); a3) *human-agent interaction* (computer-aided semiosis [12], visual ontologies). b) *Explicit non-algorithmic agent design*: b1) *emulating agent self-awareness*; b2) *e-maieutics*. (Both are referred to in detail in Section 3.)

2.4. Agent-Oriented Software

"Agent-oriented" is nowadays self-explaining, but why "software" instead of "applications" or "systems"? It expresses the perspective shift: in previous stages [9] [29] [8] the focus was on experimental models developed to validate an approach, regardless of the effectiveness of the tools involved; here, the current research stage requires focusing on software engineering too, beyond conceptual validation. However, the word "engineering" was leaved out because of two reasons: the mechanisms are described in other papers (for instance, [5]); it could have been interpreted that the mechanisms mentioned here are in a final development stage.

3. Complexity as Malady, Emergence as Antidote

After an abridged review of using emergence in stigmergic control and in agent self-awareness respectively, the focus is on trying to find a common denominator for potential leverage effects generated by any kind of emergence.

3.1. Simulated Emergence Fights Structural Complexity

Complexity has various meanings depending on the scientific field. For example "in computational complexity theory, the time complexity of a problem is the number of steps that it takes to solve an instance of the problem as a function of the size of the input […], using the most efficient algorithm. This allows to classify problems by complexity class (such as P - Polynomial time, NP - Non-deterministic Polynomial time." (http://en. wikipedia.org/wiki/Complexity). When the problem is structurally complex, carrying out simulations in a simple, safe, inexpensive, easy-to-duplicate manner is affordable and can fight structural complexity [13] [28]. Even if in ant-inspired algorithms emergence is impressive, the trouble to understand what is in fact going on at system level, is less upsetting than in the case of more familiar sub-symbolic paradigms (as artificial neural networks or evolutionary algorithms) since ant behaviour is easier to follow due to its simplicity: the ant travels from the ant hill to the food source and back guided only by pheromones. In the case of ant-inspired algorithms the solution of the problem emerges from the simulated interactions of simple entities, each with a small set of simple rules so they are very effective in dealing with problem complexity but because the system makes use of a large number of entities (see subsection 2.2) it is affordable only through simulation.

In [13] [28] are presented a series of tests aiming to determine the minimum number of "digital ants" necessary to solve, within an acceptable timespan, a nondeterministic polynomial-time hard (NP-hard) problem (in this case the Travelling Salesman Problem – TSP because it's easy to understand but hard enough to solve). The results showed that the same solution quality can be obtained with a significantly less number of ants than used in common benchmarks, saving thus at least one order of magnitude of processing time.

3.2 Emulated Emergence Fights Cognitive Complexity

"Examination of literature shows that different types of emergence exist – the self-organised structure of birds flocking is quite different from the emergence of the first self-reproducing cells – at least in terms of the creativity of the system […] Evolution depends, at least to some degree, on control of dynamics by rate-independent memory structures. These memories must first appear before the complex systems may capitalise on them." [27].

In the case of agent self-awareness this "creativity of the system" is high but vague and hard to reflect in software. Here are just some of the main reasons: a) the logics needed to formalise (self-)awareness, i.e., (auto)epistemic logics are too complex to be affordable outside large-scale systems; b) how self-awareness emerges is unknown[3]; c) the working hypotheses based on Hofstadter's ideas [22] are unproved (and practically not even tested); d) despite the high likelihood that self-reference is involved in the emergence of self-awareness, nothing proves that it is the only factor responsible; e) moreover, it is impossible to claim that agents could achieve self-awareness through Gödelian self-reference *per se* [9]; f) supposed it arises, self-awareness must be assessed, yet indirectly; if so, in evaluating a macrofeature – for instance, the ability to learn – how could be its role discriminated?; g) corollary: despite its elusiveness, agent self-awareness must be described by an unambiguous feature together with a performance metrics to assess it (a kind of surrogate of the "Travelling Salesperson Problem").

3.3. The Sigmoid Pattern: How to Exploit it

The very concept of *bounded rationality* involves suboptimality in most nontrivial applications [32] [31]. It can be easily achieved through emergence used as leverage:

Of course, any kind of emergence has lever potential but simulated emergence seems better suited to this aim. Thus, to find a common denominator for potential leverage effects, the starting pattern will be that of the solution of ant-inspired algorithms, having the following proprieties: the minimal number of digital ants necessary to get an acceptable solution is a threshold (T) that depends on problem type and complexity; the function that describes the relation between the solution quality and the number of digital ants (or the search duration) is a sigmoid (Σ); the Σ function can be modified at run-time to try various trade-offs between solution quality and time (such a sigmoid was described in [13] from the viewpoint of stigmergic control and commented upon from the viewpoint of affordability in [8]); it exhibits regularities (emerges exponentially fast as described also in [30]).

Now, widening the perspective (i.e., removing any link to the IT domain the data stem from), the sigmoid in Figure 1 should be seen as a *filtered pattern*:

- *The Pattern*. To be able to model any emergent phenomenon, the abscissa axis represents the *time* (mandatory for any process modelling), while the ordinate axis represents an undefined *solution quality* (heavily depending on the specific application). Any sigmoidal function has its shape characterised pragmatically by three attributes:
  - *Threshold* (T): determines the moment t_T when the solution quality begins to improve very fast, i.e., where self-organization becomes manifest;
  - *Cut-off point* (C): determines the moment t_C where from the solution quality will not anymore improve substantially;
  - *Steepness* (S): $S = \Delta SQ/\Delta t$. It expresses the emergence intensity: the more S tends to infinity the more quick the emergence. (From a mathematical viewpoint S expresses the closeness to the Heaviside step-function, as asymptote to sigmoid functions.)
  - In addition to the sigmoid *shape*, its *position* is paramount for investigating emergent processes, because moving the threshold towards the origin [13] means that a good solution quality emerges sooner; in Figure 1 this is symbolised by the arrow under *M* (Move).
- *The Filter*. The sigmoid as described above is irrelevant if the pattern is not used to achieve results; hence, the filter is an explicit software engineering perspective based on the modern "just

---

[3] Pessimists assume it is unknowable.

in time" paradigm (i.e., accepting suboptimality as normal, mainly under hard time constraints). The results obtained so far for simulated emergence [13] as well as the realistic expectations as regards emulated emergence [9], suggest that significant leverage is reached by moving the sigmoid and there are good reasons for expecting similar results by cutting it off. As regards the steepness, increasing it seems much less effective; exploiting the threshold needs more research.

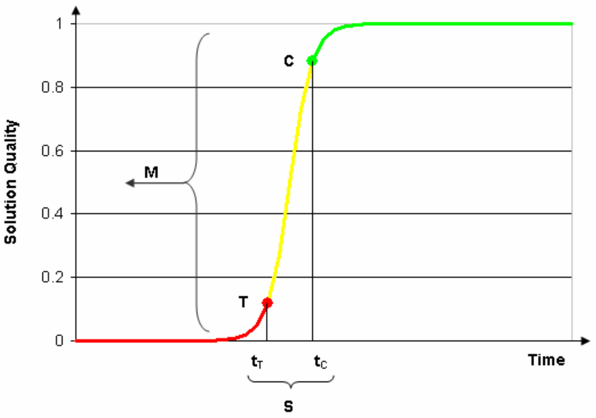

Figure 1. Emergence as leverage: solution quality versus time (how to exploit the sigmoid).

4. Misunderstood Concepts and Unaffordable or Unsuited Tools

Here AND suggests the Boolean operator to emphasize the cumulative negative effect of treating complex concepts (often ambiguous or misunderstood) with deficient (unaffordable or unsuited) tools. The problem was addressed in [10] diagnosing the main weaknesses of conventional modelling in modern AI. (They stem from inappropriate conceptualising, based on rigid, algorithmic – i.e., deterministic, almost sequential, "computational", and atemporal – processing.) To prove that the "Weltanschauung"-gap between computational-oriented mathematics and computer science is widening, just two concepts – common and widespread in both mathematics and IT – were sufficient. Thus, it was shown that *approximation* and *undecidability* refer to entirely dissimilar and incommensurable kinds of uncertain knowledge processing (approximation is deterministic and atemporal, whereas undecidability is inherently non-deterministic and has an essential temporal dimension). Though, they are sometimes considered interchangeable in dealing with another ill-applied concept: *uncertainty*. To endorse the proof given in [10], here the examples are: *uncertainty* (misunderstood concept), *logics for AI* (unaffordable tool), and *Bayesian prediction* (unsuited tool for handling the most important species of uncertainty).

4.1. Uncertainty As Epistemic Concept. Species and Degrees

From the [27], albeit quite redundant, definitions of "Uncertainty" found on the Web, only a few are interesting, since they are anthropocentric, mirroring the common user (mainly decision-maker) stance: a) "doubt: the state of being unsure of something" (wordnet.princeton.edu/perl/webwn); b) "the fundamental inability to make a deterministic prognosis" (www.kfa-juelich.de/mut/vdi/vdi_bericht_e/glossar_e.html); c) "lack of knowledge of future events" (www.projectauditors.com/Dictionary/U.html). If, as "inability" or "lack" (in its connotation of "shortage", not in that of "absence"), uncertainty could admit degrees, as "state of a decision-maker" it is obviously hard to quantify – similar to its relatives: "hope", "distrust" or "fear". Thus, all other kinds of uncertainty definitions found on the Web ("statistically defined discrepancy", "measure/degree of: variability, variety, how poorly we understand or can predict something", etc.) are very useful in many sciences (uncertainty usually appears in models as a random variable and corresponding probability density function) but hardly in typical real-world problems (for instance, economic modelling). On the other hand, uncertainty was only recently[4] accepted in the "Realm of Logic": some of the most important theories for quantifying ignorance (using probabilities or possibilities) as well as beliefs ("credal" or "pignistic"?) together with their illustration in non-monotonic logics used in agent-oriented systems, were developed in the last 30 years. Likewise, as regards agents,

---

[4] Hilbert believed that "*Für den Mathematiker gibt es kein Ignorabimus*" until Gödel broke the spell of the axiomatisation dream.

"uncertainty is accepted from the very beginning, eliminating the unacceptable closed world assumption and replacing it by certainty factors assigned to both rules and facts (for representing uncertainty the most affordable tool seems to be Stanford Algebra)" [29].

Some remarks about the epistemic status of uncertainty are relevant, because it depends on the professional background and on the task to carry out (better said, mostly on the time available to complete it). In short, uncertainty, what does it mean practically (mainly, subconsciously) for:

- Mathematicians: uncertain = *unknowable*. If it can be obtained through computation and/or deduction (no matter how complicated it is or how long it takes) it cannot be uncertain. Mathematics is atemporal, hence they can wait (nobody was fired because Fermat's Last Theorem was proved only after more than three centuries!). There are no time boundaries (if they should be, approximation methods are available to solve the problem).
- Software developers: uncertain = *undependable* (either untrustworthy or variable or unknown). There are deadlines but not critical ones (no beneficiary requests an implementation before the data base consistency was proved or restored). Time boundaries exist but they are far or can be overruled.
- End users (decision-makers): uncertain = *undecidable* (first of all for future contingents). Decisions must be made even with incomplete information. Time exists and reigns: it is real time (i.e., almost always not enough). Worse, time becomes an enemy: late reactions can kill.

In this context could be found a common denominator for a general definition of uncertainty – at least, acceptable to the three categories mentioned above? Uncertainty, in its widest sense, *comprises any unsure link in the chain of steps necessary to fulfil a task*. (Thus, if the task is, for instance, a logical problem, the link is an inference.)

4.2. Logics for Artificial Intelligence: Still a Luxury

As regards logics dealing with agent-related aspects, some recent work is fundamental: for many basic software engineering requests, Fisher's logic (more precisely, the "Temporal Development Methods for Agent-Based Systems") [18] seems for a non-specialist by far to be the most responsive and appropriate. However, most approaches ([18] included) are less applicable because they are sectorial (e.g., treating time without uncertainty or vice versa). A review of agent-oriented requirements for such logics is given in [11], where the following example regarding the need exception handling in AI logics comes from: Even primeval animals move "algorithmically" ("*if* gap *then* avoid, *else* go on) only a few steps, in very hostile environments. Moreover, reaction to stimuli cannot mean perpetual looking for the stimulus. The cardinal hindrance stems not from logic, but from the mechanisms employed: neither nature, nor technology can afford in the long run mechanisms involving large amount of testing because they are too time-consuming tools: "*if* temperature > n ^{0}C *then* alarm". The main problem is not the semantics of "*unless*", but the repeated checking of "*if*". (From this viewpoint, the semantics of "*unless*" in Reiter's default logic would be more tempting if it would be rather diachronic than synchronic – a bird *is* or *is not* a penguin but will never *become* one.) The agent is condemned to be a risk-taker, *hearing* (reactively) the environment, not *listening* (proactively) to it: the agent stops performing a task only *if* it hears the alarm bell. The point is that this "*if*" belongs to the metalanguage and does not involve thermometer reading!

Real-world problems show that the most important and ill-treated kind of uncertainty is that due to future contingents: decisions are difficult to make because *a relevant event not happened yet*, not because *a result is imprecise*. Moreover, its pragmatic corollary highlights a key aspect in decision-making: since any statement about a future event is undecidable, how to proceed in this case? Should it be predicted, circumvented, waited? No affordable AI logic gives yet a satisfactory solution, especially when combined with nontrivial temporal features.

Besides, even atemporal logics able to deal with uncertainty are avoided. For instance, a very old logic, applied now rarely (mainly in formal learning theory), is inductive logic, that extends deductive logic to less-than-certain inferences [33].

4.3. Bayesian Prediction: Still an Illusion

Definitions of "Bayesian inference" on the Web: "statistical inference in which evidence or observations are used to update or to newly infer the probability that a hypothesis may be true" (en.wikipedia.org/wiki/Bayesian_analysis); "statistical inference in which probabilities are interpreted not as frequencies or proportions or the like, but rather as degrees of belief" (en.wikipedia.org/wiki/Bayesian_inference). It is supposed to allow "to model uncertainty about the world and outcomes of interest by combining common-sense knowledge and observational evidence" (research.microsoft.com/adapt/MSBNx/msbnx/Basics_of_Bayesian_Inference.htm); "approach to statistics in which all forms of uncertainty are expressed in terms of probability" (www.cs.toronto.edu/~radford/res-bayes-ex.html).

Surely, decision-making implies *common-sense knowledge* but should it imply enough *observational evidence* (to be "statistically significant") too? Moreover, because the probability of a hypothesis depends also on the probabilities of all other n - 1 hypotheses, there are three major practical problems: a) knowing *all probabilities* involved (i.e., other ample previous statistics); b) claiming the *independence of all relations* (very hard to ascertain, primarily in diagnosis systems, where it would mean that all symptoms are totally disconnected!); c) whenever a new evidence appears, *all the probability set must be updated*.

Passing from Bayesian inference to probability theory itself – "approach to statistics in which all forms of uncertainty are expressed in terms of probability" (www.cs.toronto.edu/~radford/res-bayes-ex.html) –, the difficulty is increasing. Some of the hard questions: is the sequence of events consistent – with or without any recognizable patterns or regularities,– are long-term results in line with the law of large numbers, and, most important, are all events involved *indisputably equiprobable*? Even if decision-makers could get all that kind of answers in due time, would they believe them strongly enough to make critical decisions only on their basis? Humans are not "probabilistic beings" and are very prone to any sort of "gambler's fallacy" (the monegasque welfare is an undeniable evidence). Nothing is solved. Cui prodest?

5. Non-Algorithmic Mechanisms. Examples

The two examples were purposely chosen as unlike as possible, to illustrate the broad range of mechanisms able to support uncertain knowledge processing in agent-oriented software. (Thus, the only shared architectonic feature is their anthropocentric design [4].) The first is a straightforward interface mechanism meant for the common denominator of all kinds of uncertainty, in line with the general definition given in subsection 4.1, was implemented ten years ago and needs only the simple programming infrastructure of the traditional "Windows-programming" style of the nineties. On the contrary, the second is a complex toolbox, meant for the most demanding type of uncertainty – undecidability due to future contingents, – has only the first tool in testing and required a definitely novel approach.

5.1. Analog Input

Not only "algorithmic reasoning", but any algorithmic interaction of analog beings within analog environments is unnatural – in almost any meanings of the word [29]. While IT is now powerful enough to afford interfaces enabling users to interact with technology in their natural, ancestral, analog manner, nearly forty years of manifestly digital IT structures have induced the feeling that "digital" involves a kind of Frankenstein-like deviant and dangerous feature [7]. In addition, since the human mind is mainly visual oriented, an analog communication style seems to be more effective from the point of view of cognitive ergonomics, too.

On the other hand, by its very character, uncertainty needs analog input: beliefs[5] vary gradually and are inherently analog – their digital "percentage" expression is only an eccentric (and, often, aberrant) conversion. The features of analog data input can be revealed by ranking the information involved, in accordance to its increasing degree of certainty (somehow similar to a decreasing pragmatic significance of the proposed approach):

a) *Uncertain Problem-Solving Knowledge*. The primary source of uncertainty is the abductive reasoning unavoidable in any knowledge-based system developed for industrial or medical diagnosis purposes. Hence, some certainty factor must not be an attribute value loaded in the knowledge base any more; instead, the end user is practically able to tune the system with the mouse, adding a new capacity to the graphical interface. (A beneficial side-effect: due to the user's ability to modify dynamically the system behaviour without affecting the code such a feature is serviceable in any testbench for AI applications.)

b) *Intrinsically Fuzzy Data.* The category comprises magnitudes fundamentally unsuitable to be

---

[5] "*Belief*" means here the individual expectation regarding the likelihood of something (e.g., an event to happen, a search to succeed, a rule to be adequate).

quantified (with semantics matching groups of words as “almost sure”, “likely”, “contrary to expectation” etc.). Beside the mentioned ones, many input data – particularly in applications for industrial or medical diagnosis – are intrinsically fuzzy and to render them numeric values may be less effective or even senseless. Examples: “wear”, “maintainability”, “image sharpness”, “ruggedness”, “engine noise”, respectively “chest pain”, “pallor”, “perspiration”, “breathlessness”; if engineers or physicians were forced to evaluate them digitally, maybe they would hesitate or, at least, they will feel uncomfortable.

c) *Roughly Estimated Data.* Here are involved data concerning magnitudes measurable in essence (like “vibrations” or “weight loss”), but either assessed “organoleptically” in preliminary examinations, or approximated in average, predicted, evaluated by rule of thumb, etc.

d) *Accurate Numeric Data.* Even though it seems nonsensical, analog input of genuine digital data has three reasons: all data are input in a like manner; some basic semantic validation is performed mechanically (the user cannot input values outside the defined boundaries); it is just an alternative.

As a result, all data categories mentioned – but, first of all, data belonging to the first categories – could be input unconstrained, as analog ones, via user-friendly scrollbars, without an unnatural previous analog/digital conversion. Additional benefits are: *fine granularity* (the steps can be as little as necessary; the various degrees of accuracy and relevance of several pieces of information can be taken into account); *user-friendliness* (the control is just like any other mouse-driven activity; the data can be readily inspected as a whole); *total autonomy* (the mouse movements have prompt effect; no compilers, knowledge editors, etc. are needed: the code could be “read-only”); *robustness* (nothing essentially new, but easily implemented: all data have implicit values; user intervention is not mandatory; structured exception handling protects against input errors; the implicit semantic validation, mentioned above).

5.2. Decision-Making with Future Contingents

“The way humans make inferences proves that nature created in our brains the amazing blend of (a kind of) "von Neumann"-like algorithmic procedures (in the left hemisphere) with non-algorithmic (creative, heuristic, emerging) procedures (in the right hemisphere). As a rule, in agent strategic decision making, the layer of mental (symbolic) context should prevail over the layer of situational (sub-symbolic) context” [29]. Corollary: logic as a whole, and mainly modern agent logics could not lag behind, remaining at the semantics of pure Chrysippean logic. First came to light *dialects* of bivalent logic (e.g., modal, temporal, or non-monotonic logics). However, the Manichaean shortcomings of bivalence as such, compelled logic to create *extensions* too, becoming polyvalent: truth is not anymore perceived as atomic. The essence of many-valued logics (MVL) relevant to this paper is given in [21]: “They are similar to classical logic because they accept the principle of truth-functionality, namely, that the truth of a compound sentence is determined by the truth values of its component sentences (and so remains unaffected when one of its component sentences is replaced by another sentence with the same truth value). But they differ from classical logic by the fundamental fact that they do not restrict the number of truth values to only two: they allow for a larger set W of truth degrees. […] there does not exist a standard interpretation of the truth degrees. How they are to be understood depends on the actual field of application”. The best known many-valued logics have three, four, or infinite values (as fuzzy logic); such logics are not very popular among decision support systems (DSS) designers, since they are still a “luxury organon” for decision making because of complexity (both structural and cognitive), lack of common development environments, lack of downward compatibility, and so on. In short, despite the numerous logics created to treat undecidability in DSS, they are hard to afford in usual contexts because they are not “anthropocentric enough”.

In many-valued logics it “is general usage, however, to assume that there are two particular truth degrees, usually denoted by "0" and "1", respectively, which act like the traditional truth values "falsum" and "verum".” [21]. Obviously, any decision-making needs those pillars of bivalence. Thus, the problem is to find a truth value of “What Is in Between” corresponding semantically to the “undecidable” of a future contingent, i.e., a trivalent logic semantics (a detailed investigation is presented in [5]).

Since here the issue is to design a *mechanism* not a particular *application*, the cardinal concern, from a clear-cut software engineering perspective, is about reducing complexity, both *structural* (to make the mechanism useful to legacy systems too) and *cognitive* (to motivate system designers as well as to increase user acceptance). Hence, for the sake of simplicity, the trivalent semantics should be grounded on a usual bivalent software infrastructure. In fact, the current version of DOMINO (***D***ecision-***O***riented ***M***echanism for “***I***F” as ***N***on-deterministic ***O***perator) mechanism – described in detail in [5] – is based on common API functions callable from a customary Java development environment.

Other tools of the software engineering toolbox AGORITHM[6] (***AG**ent-**OR**iented **I**nteractive **T**ime-sensitive **H**euristic **M**echanisms*) dedicated to affordable decision making in the context of future contingents could be designed alike.

6. Conclusions and Future Work

Since the paper has a two-level target, the assessment involves both facets: A) *factual* conclusions (evaluating the non-algorithmic approaches and mechanisms proposed) and B) *general*, broad-spectrum ones (regarding emergence as leverage).

A) Factual conclusions. Since research focused on using emergence as such was recently reviewed in [28] (for stigmergic coordination) and [9] [29] (for agent self-awareness), they are not repeated here.

A1. The "Weltanschauung"-gap between computational-oriented mathematics and computer science entails the need of revisiting conventional approaches regarding the way *uncertainty* and *temporal aspects* are dealt with in agent-oriented applications.

A2. Corollary: new affordable mechanisms are needed to reduce complexity (both cognitive and structural) in AOSE.

A3. Analog input is natural (human mind is visual oriented), general (for any certainty degree), effective (fast, robust, ergonomic) and very easy to implement.

A4. AI logics are either ineffective or unaffordable; the most important and ill-treated kind of uncertainty is that due to future contingents.

A5. Bayesian inference and often probability theory itself are ill-applied (humans are not prone to probabilistic thinking).

A6. To handle undecidability due to future contingents, trivalent semantics should be based on bivalent software infrastructure.

B) General conclusions. Except the first, they are all preliminary conclusions and need further evidence to enable validation.

B1. In AOSE bounded rationality involves suboptimality. It can be achieved through emergence used as leverage.

B2. Any emergence has lever potential but simulated emergence is better suited to this aim.

B3. A pragmatic approach is based on representing solution quality as a sigmoidal function of time.

B4. The undertaking described here got significant leverage by moving the sigmoid and gives good reason for expecting similar results by cutting it off.

B5. Increasing the sigmoid steepness seems much less effective; exploiting the threshold needs more research.

Future work. The milestones for sectorial undertakings are set by four PhD theses in preparation; the closest aims are outlined in [28] [9] [16] [19].

REFERENCES

1. AgentLink III. **Agent based computing. AgentLink Roadmap: Overview and Consultation Report.** University of Southampton. http://www.agentlink.org/roadmap/al3rm.pdf, 2005.
2. AMIR, E., M.L. ANDERSON, V.K. CHAUDHRI, **Report on DARPA Workshop on Self-Aware Computer Systems.** Artificial Intelligence Center SRI International, 2004.
3. ANDERSON, M.L., D.R PERLIS, The roots of self-awareness. **Phenomenology and the Cognitive Sciences**, 4, pp. 297–333, Springer, 2005.

---

[6] In this acronym the most telling symbol is the missing "L".

4. BĂRBAT, B.E., **Communicating in the world of humans and ICTs**. Chapter 8 in L. Fortunati (Ed.) COST Action 269. e-Citizens in the Arena of Social and Political Communication, pp. 113-142, EUR21803, Office for Official Publications of the European Communities, Luxembourg, 2005.
5. BĂRBAT, B.E., **DOMINO: Trivalent Logic Semantics in Bivalent Syntax Clothes**. International Journal of Computers, Communications & Control, **2**, 4, 2007. (Forthcoming.).
6. BĂRBAT, B.E., A. MOICEANU. I, **Agent. The good, the bad and the unexpected: The user and the future of information and communication technologies** (B. Sapio et al, Eds.), Conf. Proc., Brussels, COST Action 298 Participation in the Broadband Society, CD-ROM ISBN: 5-901907-17-5, 2007.
7. BĂRBAT, B.E., A. MOICEANU, H.G.B. ANGHELESCU, **Enabling Humans to Control the Ethical Behaviour of Persuasive Agents,** Chapter 14 in E. Mante-Meijer, L. Haddon. E. Loos (Eds.) The Social Dynamics of Information and Communication Technology, Ashgate, Aldershot, UK. (To appear: February, 2008.)
8. BĂRBAT, B.E, A. MOICEANU, S. PLEŞCA, S.C. NEGULESCU, **Affordability and Paradigms in Agent-Based Systems.** (Forthcoming, in the "Computer Science Journal of Moldova".)
9. BARBAT, B.E., A. MOICEANU, I. Pah. **Gödelian Self-Reference in Agent-Oriented Software**, Proc. of the 11th WSEAS International Conference on COMPUTERS (ICCOMP '07*)* (N.E. Mastorakis et al, Eds.), 92-97, Agios Nikolaos, Crete, 2007.
10. BĂRBAT, B.E., R.S. MUNTEAN, R. FABIAN, **Approximation versus Undecidability in Economic Modelling**, Proc. of the International Workshop New approaches, Algorithms and Advanced Computational Tech-niques in Approximation Theory and its Applications (D. Simian, Ed.), 2007. (In print.)
11. BĂRBAT, B.E., S.C. NEGULESCU, **From Algorithms to (Sub-)Symbolic Inferences in Multi-Agent Systems,** International Journal of Computers, Communications & Control, **1**, 3, 5-12, 2006. (Paper selected from the Proc. of ICCCC 2006.)
12. BĂRBAT, B.E., S.C. NEGULESCU, A.E. LASCU, E.M. POPA, **Computer-Aided Semiosis. Threads, Trends**, Threats. Proc. of the 11th WSEAS International Conference on COMPUTERS (ICCOMP '07) (N.E. Mastorakis et al, Eds.), 269-274, Agios Nikolaos, Crete, 2007.
13. BĂRBAT, B.E., S.C. NEGULESCU, C.B. ZAMFIRESCU, **Human-Driven Stigmergic Control. Moving the Threshold**. Proc. of the 17th IMACS World Congress (Scientific Computation, Applied Mathematics and Simulation), (N. Simonov, Ed.), e-book, ISBN 2- 915913-02-01, Paris, 2005.
14. CORNING, P.A., **The Re-Emergence Of "Emergence": A Venerable Concept In Search Of A Theory,** Complexity **7**, 6, 18-30, Wiley Periodicals, Inc 2002.
15. DARPA. Workshop on Self-Aware Computer Systems 2004. **Statements of Position**. http://www.ihmc.us/users/ phayes/DWSAS-statements.html#top
16. FABIAN, R., S. PLEŞCA, **Stigmergy-Based Software Toolkit For Virtual Enterprises**, Proc. of the 11th WSEAS International Conference on COMPUTERS (ICCOMP '07) (N.E. Mastorakis et al, Eds.), 438-441, Agios Nikolaos, Crete, 2007.
17. FIPA TC Agent Management. FIPA Agent Management Specification. Standard SC00023K (2004/18/03). http://www.fipa.org/specs/fipa00023/SC00023K.pdf, 2004.
18. FISHER, M., **Temporal Development Methods for Agent-Based Systems**, Autonomous Agents and Multi-Agent Systems, 10, 41-66, Springer Science + Business Media Inc., 2005.
19. GEORGESCU, A., B.E. BĂRBAT, **Approximating Protensity in Computer-Aided Semiosis.** (In preparation.)
20. GOLDSTEIN, J., **Emergence as a Construct: History and Issues**, Emergence, 11, 49-72, 1999.
21. GOTTWALD, S., **Many-valued Logic**. In Stanford Encyclopedia of Philosophy (E.N. Zalta, Ed.). http://plato. stanford.edu/entries/logic-manyvalued/, 2004
22. HOFSTADTER, D.R., **GÖDEL, ESCHER, BACH: an Eternal Golden Braid**. (Including the Preface to the Twentieth-anniversary Edition.) Basic Books, New York, 1999.
23. HOLLAND, J.H., Emergence: From Chaos to Order*,* Addison-Wesley Helix, Reading, MA, 1998.

24. JOHNSON, S., **Emergence: The Connected Lives of Ants, Brains, Cities and Software**, Charles Scribner's Sons, New York, 2001.

25. LUCK, M., P. MCBURNEY, C. PRIEST, **A Manifesto for Agent Technology: Towards Next Generation Computing**, Autonomous Agents and Multi-Agent Systems, 9, 203-252, Kluwer Academic Publishers, 2004.

26. MCCARTHY, J., Notes on Self-Awareness, www-formal.stanford.edu/jmc/selfaware/selfaware.html, 2004.

27. MCDONALD, D.M, G.R.S. WEIR, **Developing a conceptual model for exploring emergence**. University of Strathclyde, 2005.

28. NEGULESCU, S.C., C.B. ZAMFIRESCU, B.E. BĂRBAT, **User-Driven Heuristics for nondeterministic problems**, Studies in Informatics and Control, **15**, 3, 289 -296. (Special issue dedicated to the 2nd Romanian-Hungarian Joint Symp. on Applied Computational Intelligence, Timişoara, 2005.), 2006.

29. PAH, I., A. MOICEANU, I. MOISIL, B.E. BĂRBAT, **Self-Referencing Agents For Inductive Non-Algorithmic e-Learning**, Proc. of the 11th WSEAS International Conference on COMPUTERS (ICCOMP '07), (N.E. Mastorakis et al, Eds.), 86-91, Agios Nikolaos, Crete, 2007.

30. PARUNAK, H.V.D., S. BRUECKNER, J.A. SAUTER, R. MATTHEWS, **Global Convergence of Local Agent Behavior**, The Fourth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'05), http://www.erim.org/~vparunak/AAMAS05Converge.pdf, 2005.

31. RUBINSETIN, A., **Modeling Bounded Rationality**, MIT Press, 1998.

32. SIMON, H.A., **Models of Bounded Rationality**, MIT Press, 1997.

33. VICKERS, J., **The Problem of Induction,** In Stanford Encyclopedia of Philosophy (E.N. Zalta, Ed.). http:// plato.stanford.edu/entries/induction-problem/, 2006.

34. ZAMBONELLI, F., A. OMICINI, **Challenges and Research Directions in Agent-Oriented Software Engineering,** Autonomous Agents and Multi-Agent Systems, 9, 253-283, Kluwer Academic Publishers, 2004.