

# Agents from Functional-Computational Perspective

**Jozef Kelemen**

Institute of Computer Science, Silesian University, Opava, Czech Republic  
VSM College of Management, Bratislava, Slovak Republic  
kelemen@fpf.slu.cz

---

*Abstract: The contribution sketches a functional-computational typological scale of agents starting from the reactive ones, and puts the family of (at least minimally) conscious agents into the proposed typology. Then it discusses the traditional computational properties of agents according their types, and sketches a way of a rather non-traditional computational characterization of conscious agents using the concept of hyper-computation. The contribution ends with relating the sketched formal approach to agents with agents embodiment, and relates embodiment of agents with their emergence of hyper-computational power.*

*Keywords: Agent, functional specification, computational specification, emergence, grammar systems, hyper-computation, consciousness*

---

## 1 Introduction

Agents might be considered from several points of view. The most usual is their constructivistic specification through their architectural modules and their mutual interrelatedness. Another usual point of view is derived from analyzing agents functional properties. The agents' functional properties are often very closely related with computational properties implemented in their architectural modules or emerging from the interactions of them. So, the functional-computational characterization of agents may contribute to our better understanding of agents. This article is devoted to such kind of characterization.

We give a stratification of agents starting from the reactive ones up to the simple emotional agents, and a way how to extend the standard typology of agents by the family of (at least minimally) conscious agents. The particular strata will be specified by most characteristic functions of agents performed in their environments or in their internal states (if any). These functions are related with the question of their effective computability, and it is hypothesized that in order to achieve (at least minimally) conscious agents, the traditionally used Turing-type

computable functions must be replaced by the so called hyper-computable functions.

By an *agent* we mean – similarly as it is almost generally accepted in present day computer science, cognitive science, and artificial intelligence; cf. e.g. (Tecuci, 1998) – any autonomously active entity with certain possibilities to sense its environment, and act in it in order to achieve certain states of this environment in which certain previously specified goals are achieved.

Agents we will propose to specify according their *functions* performed during their interactions with their outer environment on dependence of their internal states as well as these performed on their internal constituents (states, if any). These functions we will supposed (but we will not define them in this level of rigor!) as well-formalized in the sense that their precise computational analysis is possible. The most interesting property of functions supposed by us will be their *computability* in the sense of theoretical computer science. We will suppose that all the functions which specify the classes in our model (expect perhaps those specifying the conscious agents) are computable in the sense of traditional Turing-computability.

The principal common property of all of artificially designed, and more or less autonomous agents is that their parts (also in many of the cases when these agents are fully reactive) are programmed, so that they have the form of computer programs. In the theoretical level it means that these programs are equivalent with computable functions in the sense of the Turing computability. Accepting the traditional so called *Church-Turing Thesis*<sup>1</sup> this means that agents are not able to do more that the universal Turing machine can compute. In this article we will deal with different types of agents and we will argue that the computational power of certain agents may go behind the above-sketched boundary given by the Church-Turing thesis.

After a short functional stratification of agents we will concentrate to the functional specification of agents with (at least certain low level of) consciousness and we will discuss from a specific computational point of view a possibility of emergence of this kind of consciousness in agents. Our specific view will be based on considering consciousness as an emergent phenomenon appearing in agents thanks to their inner multi-agent like functional structure, and their massive interactions with outer environments. This view is in certain extent inspired by the Fodor's concept of the so called *modularity of mind* (Fodor, 1983) and by the concept of mind as a society of communicating agents in the Minsky's *society theory of mind* (Minsky, 1986).

---

<sup>1</sup> For more details on the history and formulations of the Church-Turing thesis by W. Sieg see in (Wilson, Keil, 1999, pp. 116-117).

We will formulate and analyze the question how realistic is the proposed functional specification of (conscious) agents from the computational point of view, and we will discuss in certain details a possibility of overcoming the limitations of traditional Turing-computability by a computationally perhaps exotic – the so called hyper-computational – power of (at least minimally) conscious agents.

## 2 The Traditional Functional-Computational Stratification of Agents

In this section we will give a stratification of agents based on their relatively traditional functional specification, in other words according their abilities to couple sensing with actions in a computationally realistic way, and in certain rational ways with respect the goals which agents trying to achieve during their interactions with their environments. In majority of cases mentioned below the reality of computer implementation of functions performed by the agents proves – if we accept the so called Church-Turing hypothesis – their computability in traditional sense. The agents' rationality ranges – thanks to performed computationally sound functioning of them – form very low level rationality of reactive and hysteretic agents – cf. (Kelemen, 1996) for more details about that level of rationality – up to the higher in the cases of deliberative and emotional ones.

### 2.1 Reactive Agents

The overall framework for the following stratification supposes an agent situated in its real environment (the outer – usually the physical – world with respect the agent's perspective)  $W$ , and having certain possibilities to sense this environment. Functionally, we suppose a mapping

$$\text{sensing: } W \rightarrow S,$$

where  $S$  states for the set of all data observable from the agent point of view in the environment  $W$ <sup>2</sup>. This is a very important function of agents, because connects of the real environment with the aspects of it which are accessible for the agents thanks to their sensors. Often the distinction between  $W$  and  $S$  is ignored, but in our stratification it plays some roles, so we will deal with it in the following.

---

<sup>2</sup> From the formalistic point of view  $W$  is only a symbol denoting the all possible states of agents real outer environments. Because of that the definition of the function sensing is perhaps problematic. This function is in fact defined for any agent by its technical sensors and their capacities and quality.

Another important property of any agent is the ability to act in its environment. In order to choose what to do in what situations in order to achieve the followed goal the data sensed by an agent might be associated directly (in a hard-wired manner, for instance) with certain set  $A$  of acts performable by the agent. This association might be expressed functionally as follows:

$$\text{acting: } S \times A \rightarrow W$$

Agents of the resulting functional structure

$$\text{RAG} = (W, S, A, \text{sensing}, \text{acting}),$$

where both of the functions *sensing* and *acting* are supposed to be computable in the sense of Turing-computability.

Such agents are usually, e.g. in (Arkin, 1998), called *reactive agents*. Some simple examples of such robotic agents may be found in the first part of (Brooks, 1999), for instance.

## 2.2 Hysteretic Agents

An important property of more complicated agents as the purely reactive ones is their ability to act in its environment not only on the base of sensed data, but on the base of their inner states, too. We denote the set of inner states of such agents by  $I$  in the following. All other notations remain unchanged.

Perhaps the simplest case of the use of inner states in agents functional architectures is the definition of a set of explicit goals the agents follow. Act selection then consists in associating of the just sensed data and the explicit (just followed) goal belonging to certain set of goals considered as a subset of the set  $I$  of the agent, with activity with certain element of the set  $A$  of all possible acts of this agent which it is able to perform in its environment. This coupling of sensed data and the internal state of the agents with acts performed by agents in their environment we express formally as a (computable in the traditional sense) function:

$$\text{act-selection: } S \times I \rightarrow A$$

The execution of the selected act by an agent environment means certain change of this environment caused by the agent activity in the just actual state of it. So, the acting of such type of agents will be expressed functionally as:

$$\text{acting: } S \times A \rightarrow W$$

Note that for a given particular  $s \in S$  and  $i \in I$  the *act-selection* function defines an act  $a \in A$ . The *acting* is in this particular case applied, of course, to the selected given  $s \in S$ , and to the before selected act  $a$ . So, the function act may be defined

also – and perhaps in a more precise, but from other point of view in a less readable form – also as follows:

$$acting(s, act-selection(s, i)) = w$$

where  $w \in W$ .

However, we will omit the just demonstrated way of expression of how the function appearing in agents specifications are interrelated, and we believe that it will cause no serious problems with respect of comprehensibility of the text.

Of course, the inner states of the agent may change after executing an act in the environment (by achieving a given goal another goal might become actual, for instance). This change of the agent inner state will be defined by the mapping:

$$inner-state: I \times A \rightarrow I$$

Agents of the resulting functional structure

$$HAG = (W, S, A, I, sensing, act-selection, acting, inner-state)$$

are usually, e.g. in (Genesereth, Nilsson, 1987), called *hysteretic agents*. We mention that all the functions appearing in the specification are supposed to be computable in the traditional sense.

Good examples of this type of agents are those developed in the frame of the so called new NAI (new artificial intelligence) movement of 80ties and 90ties of the past century, esp. under the intellectual influence of papers collected in (Brooks, 1999).

### 2.3 Deliberative Agents

Deliberative agents are able to select the appropriate sequences of acts for achieving their goals from their starting states as a result of certain level of their ‘reasoning’, inferring some new data on the base of sensed data and stored knowledge represented in their memories in suitable data and knowledge representation structures.

The inference process – executed usually by some *inference engine* of agents– is based on agents inner states, the sensed (and well-represented knowledge about) states of outer environment of agents, and on goal states (belonging to S, and to I, res.) of agents. This selection process is usually called *planning*, and as a part of I use well-represented *knowledge-bases* of agents. However, at least from the functional point of view, the most important functional properties of those type of agents can be expressed formally by the mapping

$$planning: I \times S \times I \rightarrow A^+$$

where  $A^+$  states for the set of all nonempty sequences of the elements of  $A$ . So, *deliberative agents* might be specified as follows:

$$\text{DAG} = (\text{W}, \text{S}, \text{A}, \text{I}, \textit{sensing}, \textit{act-selection}, \textit{acting}, \\ \textit{inner-state}, \textit{planning})$$

Good examples of deliberative agents are those developed during the research in the field of the so called GOFAI (good old-fashioned artificial intelligence) in 60ties and 70ties of the past century as presented e.g. in (Winston, 1992). Particularly the Shekey/STRIPS system (Fikes, Nilsson, 1971) is usually mentioned as a typical project of this developmental period of AI. The existence of particular deliberative agents proves again the possibility of defining the above-mentioned functions in computationally traditional ways as Turing-computable functions.

### 3 Emotional and Attentive Agents

During the 90ties of the past century, the problem of robot emotions attracted the interest of the specialists of advanced robotics, artificial intelligence, and cognitive science. M. Minsky (1986) discussed emotions in some details in the context of his society theory of mind, for instance.

Thank to the increasing professional interest, the problem has been shifted from the contemplative level to the level of laboratory design and experimentation with emotion machines. C. Breazeal developed perhaps the first robotic head – the Kismet – with the ability to achieve (and to express externally) its own inner emotional states (Breazeal, 2002).

Analyzing both of the above mentioned approaches to machine emotions from functional perspective we recognize that emotional states of the agents result from the just sensed environment, the just actual internal state of the agents, and of the (possible contradictory) goals with which agents are just confronted. C. Breazeal (2002, p. 110), introducing her robotic head, writes: *The emotions are triggered by various events that are evaluated as being of significance to the “well-being” of the robot.* So, using our formalization, we may write:

$$\textit{emotion}: I \times S \times I \times E \rightarrow E$$

where  $E$  denotes the set of all emotional states of an agent<sup>3</sup>.

---

<sup>3</sup> It is possible, of course, to suppose also the very realistic situation, that the agent's present emotional state determine in certain extent its internal and/or emotional states in the future, but we do not formalize this situation. However – because we are inspired by Breazeal's

Note two things concerning the above function: First, that we suppose two (not necessary different!) internal states, the actual state of the agent, and its goal state, and, second, that the computed emotional state of an agent depends also on the just actual emotional state of it.

Emotions play important roles in generating and optimizing plans of behaviors of real living agents (human beings) with respect of the necessity of deciding between perhaps contradictory sub-goals generated during the inference processes (and pushing agents into the states of ‘well-being’), when the necessity of conflict resolution appears during the inference processes, or during the processes of bounded rational decision-making. No matter how neutral and rational a goal may seem, it will eventually conflict with other goals if it persists for long enough. No long-term project can be carried out without some defense against competing interest, and this is likely to produce what we call emotional reactions to the conflicts that come about among our most insistent goals, writes M. Minsky (1986, p. 163). So, the question is not whether intelligent machines can have any emotions, but whether machines can be intelligent without any emotions, he continues.

The presence of emotions in the agent functional structure influence the planning process of emotional agents. This fact will be, at certain level of simplification, expressed in our formal model as follows:

$$\textit{emotional-planning}: I \times S \times I \times E \rightarrow A^+$$

Therefore, emotion agents may be characterized in the presented typology as:

$$\text{EMAG} = (\text{W, S, A, I, E, } \textit{sensing, act-selection, acting, inner-state, emotion,} \\ \textit{emotional-p planning})$$

The so called externally manifested attention – perhaps the best known example of external attention is eye movement and foveation – we will understand as a necessary restriction of the capacity of agents sensory inputs with respect their actual sensed state, and perhaps also their inner states and their emotional states. It may help agents e.g. in the process of selecting their possible actions, and may change their inner and emotional states. More formally, attention as a mapping relates to a tuple consisting of agent’s just actual observed state, its inner state, and its just actual emotional state a tuple of an observable state, an inner state, an action, and an emotional state. Formally:

$$\textit{attention}: S \times I \times E \rightarrow S \times I \times A \times E$$

The *attentive agents* are, according the above understanding of attention, described as:

---

*Kismet* as an implemented emotional agent – from the computational perspective it seems for us realistic enough to suppose the function expressing the just mentioned situation to be computationally tractable in the traditional sense.

---

ATAG = (W, S, A, I, E, *sensing, act-selection, acting,*  
*inner-state, emotion, emotional-planning, attention*)

We note that the focusing of agents attention using the above defined function may makes a change only in what the agent senses (this is the main result of focusing the attention) and on its internal and (if the agent is emotional one) also on its emotional state. The result of the focusing of attention by *attention* is then used by function participating in the process of action selection of the agent.

We suppose that the bounded rationality – in the sense of (Simon) – of some agents behaviors may be influenced (or be the result of) focusing the agents attention in the process of their inferences to the ‘important: part of their environments, an may reduce the necessary search during the inferences.

However, in present days embodied autonomous agents, e.g. in the robotic upper-torso Cog (Brooks et al., 1999), attention is usually implemented in its simplest form, only, e.g. as a mapping *attention*:  $S \times I \rightarrow S \times I \times A$ . (But for our following purposes the more general form will be more useful.) For instance, the Cog’s quite rudimental visual attention works as follows (Brooks et al., 1999): Given a visual stimulus (by waving an object in front of Cog’s cameras) it saccades to foveate on that object, and then reach out its arm toward the target. So, the result of attention focusing is the change of the robots sensed state, and it causes also some action(s) performed by the robots arm. The right arm motion is the subject of a learning process, so it s quality depends not only on the observed state of the environment, but also on the robots inner state.

## 4 Towards Conscious Agents

The main problem with the study and the attempts to model, simulate or implement consciousness in artificially created systems consist in the fact that no for scholars and for engineers have a generalz accepted meaning of the word *consciousness*. Zeman, approaching the study of consciousness from positions of a neurophysiologist with a professional recognized philosophical background, writes on three basic meenings of consciousness, all related with knowledge (Zeman, 2003): *Being awake, our first sense of consciousness, he writes, is a precondition for acquiring knowledge of all kinds. Once awake, we usually come – he continues – by knowledge through ex-perience, the second sense of consciousness. The knowledge we gain is then ‘conscious’ in the third sense we distinguished*, he completes his an-alysis (Zeman, 2002, p. 36).

From another position – from the position typical for the fields of artificial intelligence and cognitive science – is the subject of consciousness treated by P. O. Haikonen (2003). As a crutial for forming the consciousness he recognize the

phenomenon of perception of the self. *What we actually see is only the projected image on the retina, what we actually hear is the vibration of the eardrums, so why don't we just perceive these as such, percepts originating at the senses or originating at the related sensory nerve endings? How can we possibly perceive the situation being anything else?* he asks (Haikonen, 2003, p. 71).

Such questions lead to the questions on the 'internal' (mental) representation of the 'external' (physical) stimuli sensed by machines, and in consequences to the familiar mind/body problem of the philosophy of mind. His position is explained by an example (pp. 248-249): *The operation of the signals in the cognitive machine can be compared to radio transmission where a carrier signal is modulated to carry the actual radio signal. The carrier wave is what is received, yet what is detected is the modulation, the actual sound signal that is in causal connection to the original physical sound via a microphone. We do not hear the carried signal even though without it there would be no music. Thus it is possible to perceive carried information without the perception of the material basis of the carrier.*

Intuitively we also feel that for an agent to be conscious necessarily requires attentiveness and emotionality. This opinion is expressed clearly e.g. in the Aleksander and Dunmall (2003) attempt to a formal axiomatic definition of consciousness, an approach by which we will be inspired in this Section. According to the just mentioned authors, being a *conscious agent* means – intuitively and roughly speaking – to have some kind of agent's *private sense* of an outer world, of a self in this world, of self's contemplative planning of when, what and why to act, of self's own inner emotional states. Moreover it means also the conscious agent's ability to include the self's *private sense* into all of its above mentioned functional capabilities.

How to incorporate the perhaps mysterious *private sense* into the above presented spectra of functional models of different types of agents? Such an effort requires minimally an enlargement of the class of all conscious agents – say the CONAGs – by all possible variants of ATAGs, and to include the CONAG itself (at least in the form of an ATAG) into this class. In this way, some self-reference will be included into the model which will not remain without some hypotheses concerning some interesting consequences. In simplest types of agents software design, e.g. in certain HAGs, DAGs and ATAGs a very simple variant of this idea is present, however, e.g. in the form of the record of an agent's own position in its inner representation of its outer environment, e.g. in the case of robots like the MetaToto (Stein, 1994), the Shekey/STRIPS (Fikes, Nilsson, 1971), and also in the case of the case of the Cog (Brooks et al., 1999).

There exists also an opinion according to which *...consciousness will not need to be programmed in. They will emerge*, states one among the leading personalities of the present days artificial intelligence and advanced robotics in (Brooks, 1999, p. 185). Our main goal consists in the rest of this paper in arguing for the

theoretical computer science point of view for the *possibility* that consciousness *may* emerge in such a way under some circumstances (but not for *necessity* that it *will* emerge).

First of all, how to define functionally the *private sense* and how to enlarge the general functional structure of CONAGs by a suitable function? In present days mathematics it is technically extremely unusual to define functions appearing as members of their own domain of variables and domains of values. (The closest – but not identical! – concept is perhaps the more computationally oriented – so-called *recursive* – definitions of functions.)

Perhaps the simplest way how to include the simplest form of the private sense of a given particular ATAG, say B, consists in defining the mapping

$$\textit{private-sense}_B: S_B \times \{B\} \rightarrow I_B$$

and include this mapping into the functional description of this agent B which we will call a *minimally conscious agent*. So, we have a new type of agents, the type we will call MICONAG, functionally characterized as:

$$\text{MICONAG} = (W, S_{\text{MICONAG}}, A, I_{\text{MICONAG}}, E, \textit{sensing}, \textit{act-selection}, \textit{acting}, \textit{inner-state}, \textit{emotion}, \textit{emotional-planning}, \textit{attention}, \textit{private-sense}_{\text{MICONAG}})$$

Note that the MICONAG type of agents definition refers to their *'itselfs'* in some similar manner as it was in the case of the above mentioned recursive definition of the functions. However, the situation is much more complicated as in the case of simple recursive definitions because of the structure of any agent belonging into the class of the MICONAG consists not only in one mapping, but contains much more, and the 'cross-references' of functions which specifies the agent functionally, are very huge. Particularly, the (physical) environments of agents are usually noisy, dynamic, full of unpredictable changes, etc. and all these influence the behavior of agents situated in them. Because of that we may suppose that the not only the behavior of agents, but also the states of their consciousness do not be computable in the traditional sense of Turing-computability.

Nevertheless, in the following sections we show a possibility how agents constructed from simple parts, e.g. according the simple architectural principle of subsuming, may produce – under some not unrealistic conditions – more than the traditional Turing machine can compute.

## 5 Computational Analysis of the *Privat Sense*

The above mentioned experimental robots have from our perspective one important common feature: At least in certain extent, their behavioral, representational, and decision making capacities are fundamentally based on the

abilities of the present day computers to execute more or less complicated computations. In the theoretical level, these computations might be reduced into the form of the theoretical abstraction of computational processes known as Turing-computation. The Turing machine programs, with respect of their computational power in the machines over-simplified environments of symbols on a tape, and a head going one step left or right and rewriting the symbols according simple instructions sequences, differs in some senses very significantly from the real agents (e.g. embodied robots) situated in dynamically changing (data or physical) environments, and interacting with these environments very massively in many different ways. However, there exists a largely accepted hypothesis in theoretical computer science – the already mentioned Church-Turing hypothesis – according which all what is intuitively in certain sense computable (so, transformable from certain inputs into certain outputs according precisely defined and exactly executed sequences of rules – according computer programs) is computable by a Turing machine.

Especially interactions are very appealing for re-consideration of the form of ‘computation’ performed by such agents, and for drawing perhaps new boundaries between what we consider as computable and what as non-computable<sup>4</sup>. In present day theoretical computer science there are efforts to demonstrate that the notion of computation might be enlarged beyond the traditional boundaries of the Turing-computability<sup>5</sup>. In (Burgin, Klinger, 2004) is proposed to call algorithms and automata that are more powerful than Turing machines as *super-recursive*, and computations that cannot be realized or simulated by Turing machines as *hyper-computations*.

To have the *private sense* as sketched in the previous section means – metaphorically speaking – to have an ability of a given agent AG to consider itself as another agent identical with AG, and to consider this type of ‘schizophrenia’ in the work of other functions which characterize the agent AG. This type of recursion is too complicated for expressing it in the frame of the traditional paradigm of one-processor computation. It requires at least some suitable framework for dealing with behaviors that appear thanks to interrelations between individually autonomous entities (agents).

The appearing situation insinuates the framework of considering a conscious agent as a system consisting in more then one agent, so in a form of a multi-agent systems. Might be that the phenomenon of consciousness *emerges*<sup>6</sup> from

---

<sup>4</sup> Cf. (Wegner, 1997).

<sup>5</sup> For more details on the effort, see e.g. (Eberbach, Wegner, 2003) or the monothematic issue of the *Theoretical Computer Science* **317**, No. 1-3 (2004) 1-269.

<sup>6</sup> Whether the nature of consciousness is emergent or not, it is an actual question not of the present day cognitive science only, but also of the philosophy of mind. Freeman (2001) and Holland (2003) provides the overview of different opinions present in both of the fields.

interactions of such virtual agents representing of ourselves and from representations of other agents in our individual minds. The function private sense mentioned in the previous section is perhaps also an emerging product of interactions of several other functions. Inspired by Fodor's view (Fodor, 1983) we may suppose a conscious agent as structured into certain simpler 'modules' or – inspired by views of M. Minsky (Minsky, 1986) – formed as a society of simpler, unconscious and massively interacting 'agents'. Perhaps the conscious behavior of such an agent might be then described as a phenomenon, which emerges – in the sense proposed in (Holland, 1998)<sup>7</sup> – from interactions of traditionally computable behaviors of simpler constituting parts of it, and has the form of a hyper-computation. Let us try to demonstrate in the next section how it is possible to proceed in this way.

In the following Section we will try to connect the above mentioned computational problems of conscious agents with one particular formal – in certain extent rule-based – model of hyper-computations, and illustrate that from simple, massively interacting components may emerge the hyper-computational power if relatively slightly imaginable influences of real physical environments of agents are in some way taken into consideration.

## 6 Increasing Computational Power

In this Section we will illustrate that there exists at least one rigorous formal model of agents with components producing a rule-governed Turing-computable behaviors each, but producing – considered as a whole – a behavior which does not be generated traditionally by any Turing-equivalent generative device, so which requires the generative power of hyper-computation. We will consider in this role the so-called eco-grammar systems. First, we introduce in a few words this model, presented originally in (Csuhaaj-Varjú et al., 1997).

An *eco-grammar system* (or an EG system for short) consists of several abstract, formally specified autonomous entities called components. *Components* are described by strings of symbols (with the capability to developing according precisely defined so called Lindenmayer-type, or L- rules) acting on their *inner environment*, by pure rewriting of symbols in this string-like *outer environment* using precisely defined rules applied sequentially. The outer environment

---

<sup>7</sup> In (Holland, 1998, pp. 121-122) emergence is explained as '... about all a product of coupled, context-dependent interactions. Technically these interactions, and the resulting system, are nonlinear: The behavior of the overall system cannot be obtained by summing the behaviors of its constituent parts... However, we can reduce the behavior of the whole to the lawful behavior of its parts, if we take nonlinear interactions into account' (emphasized by J. H. Holland).

described by a string of symbols may also develop according to a set of L-rules. The rules used by each component for development depend on the state of its inner environment, and on the state of its outer environment. The rules used for acting in the environment depend on the state of the components.

More formally – according (Csuhaaj-Varjú et al., 1997) – an *eco-grammar system*  $\Sigma$  consists, roughly speaking, of

- a finite alphabet  $V$ ,
- a fixed number (say  $n$ ) of components evolving according sets of rules  $P_1, P_2, \dots, P_n$  applied in a parallel way as it is usual in L-systems (Rozenberg, Salomaa, 1980), and of
- an environment of the form of a finite string over  $V$  (the states of the environment are described by strings of symbols  $w_E$ , the initial one by  $w_0$ ).
- the functions  $\varphi$  and  $\psi$  which define the influence of the environment and the influence of other components, respectively, to the components (these functions will be supposed in the following as playing no roles, and will not be considered in the model of eco-grammar systems as treated in this article).

The rules of components depend, in general, on the state (on the just existing form of the string) of the environment. The particular components act in the commonly shared environment by sets of sequential rewriting rules  $R_1, R_2, \dots, R_n$ . The environment itself evolves according a set  $P_E$  of rewriting rules applied in parallel as in L systems.<sup>8</sup>

The evolution rules of the environment are independent on components' states and of the state of the environment itself. The components' actions have priority over the evolution rules of the environment. In a given time unit, exactly those symbols of the environment that are not affected by the action of any agent are rewritten.

In the EG-systems we assume the existence of the so-called *universal clock* that marks time units, the same for all components and for the environment, and according to which the evolution of the components and of the environment is considered.

In (Csuhaaj-Varjú, Kelemenová, 1998) a special variant of EG-systems have been proposed in which components are grouped into subsets of the set of all components – into the so-called *teams* – with fixed number of members. The idea was to express in the model the embodiment (the technical side) of components in certain computationally tractable form. Especially in this case through limitation of activities of components by some physical constrains, for instance.

In (Wätjen, 2003) a variant of EG-systems without internal states of components is studied. The fixed number of members proposed in (Csuhaaj-Varjú,

---

<sup>8</sup> So, the triplet  $(V, P_E, w_E)$  is (and works as) a Lindenmayer-system.

Kelemenová, 1998) is, however, replaced by a dynamically changing number of components in teams.

As the mechanism of reconfiguration, a function, say  $f$ , is defined on the set  $N$  of integers with values in the set  $\{0, 1, 2, \dots, n\}$  (where  $n$  is the number of components in the corresponding EG-system) in order to define the number of components in teams. For the  $i$ -th step of the work of the given EG-system, the function  $f$  relates a number  $f(i) \in \{0, 1, 2, \dots, n\}$ . The subset of the set of all components of thus EG-system of the cardinality  $f(i)$  is then selected for executing the next derivation step of the EG system working with Wätjen-type teams.

Wätjen (2003) proved, roughly speaking, that there exist EG-systems such that if  $f$  is (in the traditional sense) non-recursive function, then the corresponding EG-system generates a non-recursive (in fact a super-recursive) language.

Whether or not has the computational power of EG systems received in the above described way really emerges from the recursive computations of suitable configurations of its components we may test using the test of emergence proposed in (Ronald et al., 1999). The test offers an operant definition of emergence (esp. for phenomena appearing in the experiments of Artificial Life, but we consider its validity also for the study of agents and multi-agent systems). The requirements putted onto systems in which the emergence of some phenomenon appears are the following (Roland et al., 1999):

*Design.* The designer designs the systems by describing *local* interactions between components in a language  $L_1$ .

*Observations.* The observer describes global behaviors of the running system using a language  $L_2$ .

*Surprise.* The language of design  $L_1$  and the language of observation  $L_2$  are distinct, and the causal link between the elementary interactions programmed in  $L_1$  and the observations observed in  $L_2$  is non-obvious.

The emergent nature of the behavior (language) generated by the above described EG system is clear, because of the components of the given EG system generate recursive languages each, the local interactions of the components are given only and, surprisingly the whole system generates a non-recursive language (behavior).

From the technical point of view some surprising properties of components – e.g. their consciousness – may emerge from their simple parts and from interactions of these parts instead of being implemented in certain clever way into the systems. This idea may be supported not only by some theoretical considerations (or speculations) but also by the feeling of those who construct more or less intelligent robots. To illustrate that we quote R. Brooks again: *Thought and consciousness will not need to be programmed in. They will emerge* (Brooks, 1999, p. 185).

## 7 Embodiment and Hyper-Computation

The above-sketched Wätjen's results might be interpreted in the context proposed in this contribution for characterization of agents as a proof of the following statement:

An agent considered as a multi-agent system in the meaning as used in the theory of eco-grammar systems where the changing number of sub-agents which in a given moment actively participate in the generation of the behavior of the whole system

- might be described as consisting of rule-governed parts (modules or simpler agents) with abilities to perform traditional Turing-type computations, and
- is able to produce a behavior beyond the limits of traditional computability which seems to be necessary for appearing of the phenomenon of the agent's consciousness.

One among the most important achievement during the development of Artificial Intelligence (AI) was the discovery of the methodologically new possibility how to test our hypotheses on how (some of) the intellectual processes run. The history of AI is full of different hypotheses on how to 'automate' processes like general problem solving, theorem proving, natural language understanding and communication, diagnostics, image processing and recognition, scene analysis, etc. in order to obtain working computer-based systems performing these tasks at the similar (or at better) qualitative level as (specially trained) human beings perform them. In all these cases

- (1) a working hypothesis is produced first – in the majority of the cases it is based on author's own introspection, then
- (2) the formulated hypothesis is implemented (often using a suitable programming language that might be developed for such purposes), and
- (3) the developed system of programs (the implemented version of the hypothesis) is then tested on real (or more or less similar to the real ones) data.

To proceed according such methodological guidance seems to us as something natural. It might be because intellectually we feel prepared for contemplations about our own intellectual capacities.

The situation is completely different in the cases when the agents (intended to be intelligent in certain sense, e.g. cognitive robots) are situated and execute tasks in real physical environments. In such a case the systems are faced with physically grounded ontologies of objects with real physical properties that exist and act in real time scales. Very hard problems appearing in such situations in the traditional AI were pointed out first from very different positions and with very different conclusions by M. Minsky, cf. e.g. (1986), and R. Brooks, cf. e.g. (1999).

Brooks in his concept of the novel AI emphasizes the principal role of systems reactivity, which is necessary for their low-level rationality, while Minsky emphasizes the principle of decentralization and organization of simplest units (agents) into more complex ones (agencies) and presupposes that an agency may play the role of an agent in a more complex agency. Both of these positions might be – according to our conviction – combined into one unified approach. The main idea consists in two basic steps:

- (1) to emphasize the role of as direct as possible interaction of the cognitive systems with their environments at least at the lowest level of sensing and acting, and
- (2) to exploit the power of organization and of the emergence in highest levels in order to receive more complex behaviors.

Both of the above mentioned steps lead us to realize the principal difference between *implementation* of our ideas on how cognitive processes run in natural systems and how they may run in artificial ones, into more or less traditional but in certain sense rigid computers usually equipped with suitable input-output devices which isolate them from their environments by providing data from it for them, and between *embodiment* of our ideas into artificially created agents.

Realize now that our physically embodied and working in real physical environments agents are constructed from physically engineered and constructed (it means functionally not completely reliable) parts like sensors providing signals for them, units for processing signals and perhaps compute the decisions, actuators for making changes in their environments, and situated and working continuously in real, dynamic, and noisy environments. Realizing that the example and the theoretical result proved on the mathematically sound model of agents shown in the previous Section illustrate the possible influence of different physical (so, related with the ‘body’ of agents) states of components of agents into their computational properties.

Supposing a technically relatively acceptable situation – very well known for all those who do experiments with real embodied robots, for instance – that the components of robots are unreliable in certain level, we may interpret the Wätjen’s model of EG systems as an acceptable model of agents with very simple architecture (reflected in very simple communication between parts forming a whole agent) and having a hyper-computational power. This computational power is perhaps the necessary force, which pushes agents from the traditional Turing-computable behaviors toward their consciousness.

## **Conclusions**

Concluding the above notes and positions we state the following: It seems to be realistic that the deep philosophical and ethical question ‘To have conscious agents or not?’ is reducible to the much more technical question ‘To have much more complicated robots as we have now, or not?’ We have demonstrated in a rather

formal way that the consciousness of agents *may* emerge as R. Brooks has supposed that.

### Acknowledgement

The present work was supported by the grant MSM 4781305903. The author's research on the topic is partially supported by the Czech Science Foundation Grant No. 201/04/0528, and by Gratex International, Inc., Bratislava, Slovakia.

### References

- [1] Aleksander, I., Dunmall, B.: Axioms and Tests for the Presence of Minimal Consciousness in Agents. In: *Machine Consciousness* (O. Holland, Ed.). Imprint Academic, Exeter, 2003, pp. 7-18
- [2] Arkin, R. C.: *Behavior-Based Robotics*. The MIT Press, Cambridge, Mass., 1998
- [3] Breazeal, C.: *Designing Sociable Robots*. The MIT Press, Cambridge, Mass., 2002
- [4] Brooks, R. A.: *Cambrian Intelligence*. The MIT Press, Cambridge, Mass., 1999
- [5] Brooks, R. A., Breazeal, C., Marjanovic, M., Scassellati, B., Williamson, M. M.: The Cog Project – Building a Humanoid Robot. In: *Computation for Metaphors, Analogy, and Agents* (C. Nehaniv, Ed.). Springer, Berlin, 1999, pp. 52-87
- [7] Burgin, M., Klinger, A.: Three Aspects of Super-Recursive Algorithms and Hyper-Computation or Finding Black Swans. *Theoretical Computer Science* **317** (2004) 1-11
- [8] Csuhanj-Varjú, E., Dassow, J., Kelemen, J., Paun, Gh.: *Grammar Systems*. Gordon and Breach, Yverdon, 1994
- [9] Csuhanj-Varjú, E., Kelemen, J., Kelemenová, A., Paun, Gh.: Eco-Grammar Systems – a Grammatical Framework for Lifelike Interactions. *Artificial Life* **3** (1997) 1-28
- [10] Csuhanj-Varjú, E., Kelemenová, A.: Team Behaviour in Eco-Grammar Systems. *Theoretical Computer Science* **209** (1998) 213-224
- [11] Eberbach, E., Wegner, P.: Beyond Turing Machines. *Bulletin of the EATCS* **81** (2003) 279-304
- [12] Fikes, R. E., Nilsson, N. J.: STRIPS – a New Approach to the Application of Theorem Proving in Problem Solving. *Artificial Intelligence* **2** (1971) 189-208
- [13] Fodor, J.: *The Modularity of Mind*. The MIT Press, Cambridge, Mass., 1983

- 
- [14] Freeman, A. (Ed.): *The Emergence of Consciousness*. Imprint Academic. Thorverton, 2001
- [15] Genesereth, M. R., Nilsson, N. J.: *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann, Los Altos, Cal., 1987
- [16] Haikonen, P. O.: *The Cognitive Approach to Conscious Machines*. Imprint Academic, Exeter, 2003
- [17] Holland, J. H.: *Emergence*. Addison-Wesley, Reading, Mass., 1998
- [18] Holland, O. (Ed.): *Machine Consciousness*. Imprint Academic. Exeter, 2003
- [19] Kelemen, J.: A Note on Achieving Low-Level Rationality from Pure Reactivity. *Journal of Experimental and Theoretical Artificial Intelligence* **8** (1996) 121-127
- [20] Kelemen, J.: May Embodiment Cause Hyper-Computation? In: *Advances in Artificial Life* (eds. M. S. Capcarrère et al.) Springer, Berlin, 2005, pp. 31-36
- [21] Minsky, M.: *The Society of Mind*. Simon and Schuster, New York, 1986
- [22] Ronald, E. M. A., Sipper, M., Capcarrère, M. S.: Design, Observation, Surprise! A test of Emergence. *Artificial Life* **5** (1999) 225-239
- [23] Rozenberg, G., Salomaa, A.: *Theory of L-Systems*. Academic Press, New York, 1980
- [24] Simon, H. A.: *The Science of Artificial, 2<sup>nd</sup> Edition*. The MIT Press, Cambridge, Mass., 1982
- [25] Stein, L. A.: Imagination and Situated Cognition. *Journal of Experimental and Theoretical Artificial Intelligence* **6** (1994) 393-407
- [26] Tecuci, Gh.: *Building Intelligent Agents*. Academic Press, San Diego, Cal., 1998
- [27] Wätjen, D.: Function-Dependent Teams in Eco-Grammar Systems. *Theoretical Computer Science* **306** (2003) 39-53
- [28] Wegner, P.: Why Interaction is more Powerful than Algorithms. *Communications of the ACM* **40** (1997) No. 5, 81-91
- [29] Wilson, R. A., Keil, F. C. (eds.): *The MIT Encyclopedia of the Cognitive Sciences*. The MIT Press, Cambridge, Mass., 1999
- [30] Winston, P. H.: *Artificial Intelligence, 3<sup>rd</sup> Edition*. Addison-Wesley, Reading, Mass., 1992
- [31] Zeman, A.: *Consciousness – A User’s Guide*. Yale University Press, New Haven, 2002