Proceedings of the Towards a Comprehensive Intelligence Test (TCIT): Reconsidering the Turing Test for the 21st Century Symposium,
Aladdin Ayesh, Mark Bishop, Luciano Floridi, and Kevin Warwick (Ed.), at the AISB 2010 convention, 29 March – 1 April 2010, De Montfort University, Leicester, UK

Connecting the dots my own way: Sphex-test and flexibility in artificial cognitive agents.

Juan Camilo Espejo-Serna[1]

Abstract. The purpose of this paper is to raise doubts on the standard approach in cognitive science regarding the design of artificial intelligence. An intelligence criterion, different from the Turing test, is presented and explained: the sphex-test. Following the explanation of the test, a theoretical approach of the normative constraints of experimental setups is presented and the causes of failure determined through it. It is shown that domains with which the agent can only relate in one unique way (have only one presentation mode) are responsible for failure to pass the test. Finally, a possible solution to the problem is suggested by sketching one way of obtaining the necessary flexibility to not to fail the sphex-test.

From now on, I'll connect the dots my **own** way.
Calvin, from *Calvin & Hobbes*

Cognitive science attempts to give an explanation of what intelligence is and how organisms attain it. At first sight, the design of an artificial agent able to pass the Turing test could give us the key of this enterprise. The goal of the Turing test is to offer a criterion to determine if machines can think. Thus, an artificial agent that passes the Turing test would be legitimately attributed cognitive abilities, *i.e.*, we would attribute intelligence to it. Even if the Turing test is a good criterion for the attribution of intelligence, it only presents a sufficient condition for intelligence: an agent could fail to pass the Turing test and still have some kind of intelligence. In this sense, the Turing test is too strong: the failure to pass the test cannot guarantee the lack of cognitive abilities whatsoever. We need a test to determine intelligence such that, failure to pass it consists a proof of the lack of intelligence, *i.e.*, a necessary condition for passing the test is the possession of intelligence.

The question of what intelligence is and how organisms attain it can be reformulated as follows: what is the simplest activity system which is such that it requires intelligence to perform well at it? The standard assumption in AI in order to deal with this question is to pick a domain for which cognitive abilities are needed to perform adequately and design an artificial agent that performs adequately in such domain, and from there determine whether the artificial agent has certain cognitive abilities, and whether its architecture provides an explanation of how such cognitive abilities work.[2] In this paper I want to question this standard assumption. I will present a simple test —the sphex-test— that any artificial agent under the described approach should pass, and apply it to a paradigmatic project in the new artificial intelligence: Rodney Brooks' *creatures* (2; 3). Based on the sphex-test, I will suggest that flexibility is a necessary condition for the attribution of cognitive abilities to artificial agents, and so that the evaluation of an agent performance in a domain is not enough. I will then focus on finding a common denominator that can explain failures to pass the test, and argue that in order to ascribe cognitive abilities to an agent it is necessary to consider the presentation mode of the domain it is performing in. The complexity of proper performance in a domain changes according to the way the domain is presented. Different presentation modes ask for different behaviors of different complexity; such that these three elements (the domain, the performance and the presentation mode) are necessary for proper cognitive abilities ascription.

A general characterization of some normative constraints (bounding, constitutive and governing) is used to pin point the deficiencies and detect a particular kind of domain guilty of them. The problem is found in cases where there is a TICTACness domain: a case when there is no space for difference between the bounding normativity and the constitutive normativity. Once the problem is detected, a possible way out is presented: a domain where the presentation mode from which the agent starts has no direct relation with the correct performance in the domain. Artificial agents should behave as Calvin does: when presented with a domain, they should built up their own way of dealing with it, not restraining themselves to one way the domain is presented. Cognition is not seen when the agents perform just as the domain is initially presented: by simply connecting the dots in the predetermined manner. Artificial agents must be able to connect the dots on their own, thus truly exhibiting cognition.

1 Sphex-test and flexibility

I want to propose a simple test for any particular cognitive science project that attempts to show something about cognition through the behavior of artificial agents. I wish to test the correctness of such inferences. The idea is to *slightly* modify the environment in which the agent performs and examine if the original claims made about the agent still hold. A change in the environment should only cause a modification of the agent's performance (*v.g.*, render it not as well-adjusted) but should not modify the kind of behavior, *i.e.*, it has to be possible to describe the agent's behavior in the same way albeit with small qualifications. For instance, if an artificial agent is described by its designers as exemplifying a plan-following behavior inside a given environment, it should also be possible to describe that same agent as following plans in the modified environment (but perhaps in a sloppier manner).

Let us take the example of the digger wasp (wasps of the genus *Sphex*). Inside its niche the digger wasp can be said to have a plan-following behavior: it "wants" to take care of its eggs and has a plan

[1] National University of Colombia. Email: jcespejos@unal.edu.co

[2] Exactly this is Shannon's argument for attempting to build a computer that can play chess like a master: if playing chess like a master requires intelligence and a computer manages such skill of game play, then such computer would be intelligent, see (9).

for it. The wasp looks for a prey, stuns it, digs a burrow, hides the prey inside and lays its eggs besides it so that the newly born larvae will have food right away. If the original description of the wasp's behavior as plan-following was correct, then an alteration in the environment would not change it radically. (Although —we would imagine— the wasp would have to modify its original plan in order to cope with the changes). This situation is presented by Woodridge (11), where the wasp's catch-dig-bury procedure is interrupted by an experimenter that slightly changes the position of the stunned prey while the wasp is checking that the dug burrow has nothing unexpected (another predator ready to eat both eggs and prey, for example).[3] If the description of the wasp's behavior as plan-following was correct, then in this modified environment the wasp would still develop a plan. Yet the wasp cannot handle the modification. As the wasp goes inside the burrow and the prey is moved by the experimenter, and then the wasp comes out, it returns the prey to its original position and reexamines the burrow. This is repeated *each* time the prey is moved. The experimenters moved the prey up to forty times and the wasp kept on repeating the same action all over again. Not a single time did the wasp ignore the change and move on. The initial cognitive explanation for the behavior of the wasp was proven incorrect by a simple modification: the allegedly plan-following behavior of the digger wasp was revealed as absurd repetition.

The wasp, that in one scenario behaved as if it had cognitive abilities, behaves as lacking such abilities in the modified scenario: in one case behaved as if it could follow plans, while in the modified case it behaves as repeating an absurd routine. If we are going to ascribe cognitive abilities to an agent, it —unlike the wasp— should be able to perform similarly in a different environment. If an agent really has the purported cognitive abilities it should be able to exercise such abilities in more than a limited context. Based upon this we can construct a test for artificial agent: if an artificial agent fails it, the cognitive theory sustained by it would be proven incorrect. Through this test we will be able to distinguish cognitive descriptions *proper* of an agent from cognitive descriptions *read into* an agent. A cognitive description proper of an agent will survive the sphex-test, while cognitive descriptions read into the agent will fail to pass the test. This will be called the sphex-test:

> ***Sphex-test***: If it is possible to introduce a small modification in the environment that causes a massive collapse in the appearance of the alleged cognitive abilities, then such abilities are only read into the agent.[4]

In order to better understand the basics of the test let us apply it to a paradigmatic example of the new AI: Rodney Brooks' *creatures* (2; 3). These robots are designed to perform fluently in a natural environment: not a controlled experimental setup but rather a real world environment. The subsumption architecture behind them allows layers with different functions to work in parallel, such that the creatures cope appropriately and in a timely fashion with changes in its dynamic environment. Minor changes in the properties of the world should not lead to total collapse of their behavior, they can both adapt to surroundings and capitalize on fortuitous circumstances and have some purpose in being (4, 4). One of his better known robots is Herbert, designed to pick up empty soda cans around the MIT labs, from which he says has the cognitive ability to move around picking up empty cans. It has obstacle avoidance, wall following, object recognition (empty soda cans and desks) and other behaviors that allow it to search, reach and pick up soda cans (3). Herbert is designed to function properly inside the MIT lab, where the doors, walls, desks, *etc.*, look a certain way. Although perhaps not in an explicit way, Herbert is designed to move around the kind of objects encountered in the MIT lab, those that during debugging forced adjustments. If Herbert was transfered to another lab, its performance would probably lower or fail because its basic layers are not adjusted to the new environment. For example, if the MIT starts buying different sodas (with bigger and heavier cans) Herbert would leave them alone regarding them as not empty. (If the new empty soda cans weight the same as the old cans that still have some soda, Herbert will not pick up those cans.) If the sphex-test is applied to Herbert at this point, it would fail: a small change in the world (the change of weight of the cans) would make it fail in its task (pick up the empty cans).

One might say that this failure does not make Herbert collapse altogether because Brooks' crew could redesign it, as part of the debugging procedures, allowing it to handle the new kind of cans. If the cans are not being detected by the sensory apparatus, they will be modified in order to allow Herbert to behave suitably again. But such changes would be *ad hoc* and offer no guarantee of flexibility for future cases. Even more, if an external agent is the one that makes the adjustments, then a second order sphex-test can be formulated. Can the agent that modifies Herbert respond flexibly to changes in the conditions to which it must adjust Herbert? If the answer to this question is negative, then Herbert cannot pass the sphex-test because there might be a situation for which both Herbert and the engineer are not prepared. But if the answer is positive, the mechanism that accounts for this flexibility will also be responsible for Herbert's flexibility, and thus the cognitive abilities should be ascribed to such mechanism and not to Herbert. Eitherway, there are no grounds of the ascription of cognitive abilities to Herbert. If there is no way to guarantee flexibility, the correctness of the artificial intelligence enterprise is undermined. Without flexibility there is no way to be sure that the cognitive abilities the designers want to attribute to the machine are proper or just read into it and thus no legitimate way of extracting cognitive conclusions from physical models. Without an answer, there are only mechanichal systems with rather complicated behaviors. It seems then that flexibility is the mark of cognition.

The sphex-test is not meant to prove that a given agent has no cognition whatsoever. Someone could argue that, in a sense, even an ordinary thermostat has at least some kind of cognition: it has an access to the world and produces changes accordingly. This implies that to conclusively determine that an agent has no cognition asks for a detailed examination of what cognition is and how can it be exemplified: the necessary conditions of any kind of cognition should be examined. The sphex-test has a more modest reach. It aims to determine the correctness of cognitive abilities *ascriptions* to artificial agents based on a particular behavior. The digger wasp could be an intelligent agent wishing to conceal its abilities by acting stupidly, but the behavior exhibited cannot justify the ascription of such abilities. Likewise, an artificial agent that fails the sphex-test could have some kind of cognition (at least in the weak sense of the thermostat) but lacks that that was supposed to have. Cognitive scientists attempt to justify cognitive abilities ascriptions to an artificial agent because it behaves in a certain manner in a domain. The sphex-test undermines the possibility of such inference by presenting cases in which

[3] The idea of using the case of the digger wasp to examplain cases of failures or breakdowns in intelligence is from Adrian Cussins. For example, see (5).

[4] The reader is asked to accept for now this intuitive understanding of the sphex-test, in particular what regards the conditions for a change in the domain to be a "small modification". The small change must be such that the agent can still perform (perhaps badly) whatever it was doing before the change. The characterization offered in the following section should help settle possible doubts regarding particularities of the test.

Proceedings of the Towards a Comprehensive Intelligence Test (TCIT): Reconsidering the Turing Test for the 21st Century Symposium,
Aladdin Ayesh, Mark Bishop, Luciano Floridi, and Kevin Warwick (Ed.), at the AISB 2010 convention, 29 March – 1 April 2010, De Montfort University, Leicester, UK

an agent that has proper performance in a domain fails to exhibit intelligence when faced with a small change, thus suggesting that there is more than these two aspects involved, *i.e.*, other elements besides the domain and the behavior are at issue.

2 Appropriate domains

As we have seen in the previous section, there are reasons to doubt about the possibility of obtaining conclusions from the behavior of artificial agents in domains. The sphex-test suggests that there is another factor in play here, that not just any task and not just any behavior can have explanatory impact and elicit conclusions about the nature of cognition. This asks for a general characterization of which domains and which behaviors do not have explanatory impact and why, and which kind of domain and behavior, if any, could have such impact. In this section it will be claimed that a TICTACness constitution of a task domain is the reason of failure to pass the sphex-test, and so artificial agents must not have TICTACness domains. In order to do so, a general way to characterize tasks and behaviors in terms of their normative constraints will be presented and through this pin point a particular kind of constitution —TICTACness— that is common to all projects that fail to pass the sphex-test.

2.1 Domain, representation and valuation

The standard approach in cognitive science is to construct an artificial agent that performs adequately in a domain for which certain cognitive abilities are necessary. The domain, then, has to be such that cognitive abilities (and not just causally explained relations) are needed for proper performance, *i.e.*, the domain and the performance have to be a certain way. These aspects (the domain and the kind of performance required) directly determine if cognitive abilities are needed, and thus if an artificial agent performing adequately has them.

Think about tic-tac-toe (TTT). TTT is a two-player game generally played in a nine slot board, where players take turns placing different tokens in it. There are three possible board states that end the game: when the first player's tokens are in a δ combination, when the second player's tokens are in a δ combination, or when there are no more empty slots in the board. These rules and an explicit characterization of what a δ combination is delimit the *domain* of TTT, and consitute the minimum amount of information a player without any previous particular knowledge about the game must have in order to start playing. Yet this information can be *represented* in a manyfold of ways (have different modes of presentation).[5]

There is an intricate connection between the representation and the possibility of achieving a δ combination. For a regular adult, playing TTT in a 3×3 grid would be much more easier than playing TTT using another presentation of the domain[6]. Thus, the particular way of representing the domain has a role in the determination of the kind of performance within the domain. The domain can remain the same but a change in the way it is represented could change the transformations needed for proper performance.

It is not enough to examine the performance in a domain in order to ascribe cognitive abilities to an artificial agent. The particular way the domain is presented (the representation the agent has of the domain) plays a major role. Projects in cognitive science must be examined regarding the domain the agent performs, how the agent performs *and* the representation of the domain used to achieve such performance. These three factors will determine the validity of cognitive abilities ascriptions.

Remember the case of Herbert. The *domain* Herbert is designed to work in is the MIT labs. If Herbert is going to do any task (correctly or incorrectly), it must be able to comply to the general characteristics of the MIT labs. One way this domain could be presented would be in terms of objects and their properties, *i.e.*, in terms of walls, corridors, people moving around, laptops, sodas, *etc.*. Achieving the goal of picking up empty soda cans in terms of this would require having a model of the MIT labs such that a trajectory evading obstacles and recognizing and picking up cans is made possible. Yet this would need the computation of all the changes in the environment that could produce a modification in the model of the world, adjusting the model to them, and then deciding a new trajectory for the current state of the world. A feat of huge computation powers. But Herbert does not behave in this way. It deals with its domains directly as it is presented by its sensory channels, without any further processing or modelling. The domain is presented only as sensory data that causes particular movements. The subsumption architecture of layers deals with the different stimuli producing a combination of movements that result in a particular behavior that fulfils its goal. As in the TTT case, there is a way of distinguishing three determinations: Herbert's domain (the MIT labs), the way the domain is presented to it (sensory information) and its performance.

These examples present two different phenomena characterized in terms of the boundaries of a domain, its presentation and what proper performance is. This difference can be characterized as the difference between aspects that *delimit* the domain, aspects that *constitute* the domain and the *valuations* of the elements in the domain. We will call these normative aspects bounding normativities, constitutive normativities and governing normativities, respectively. The *bounding normativity* delimits the domain: what belongs and what does not to the domain; it is the one in charge of delimiting the space of the domain. The *constitutive normativity* characterizes the elements of the domain: the way the domain is presented. The same domain (the labs) can be presented in terms of objects and their properties or as stimuli and this normative constraint regulate which way and how the domain is presented. And lastly the *governing normativity* is the standard against which it is determined what correct performance in the domain is (closeness to a goal).[7]

2.2 TICTACness and failures to pass the sphex-test

Let us take now the understanding of TTT in terms of the three kinds of normativities and explain why it is prone to produce inflexibility cases. Imagine a computer program that plays TTT like a master (it plays just like any regular adult would play).[8] Yet, if this is true and

[5] For example, it can be presented as a game played in a 3×3 grid where the δ combination is made by placing three in a line or like a game played with nine different chips where the δ combination is made by a particular combination of chips. What is important is that there is a method to translate one way of presenting the game to the other, thus warrant the sameness of the game.

[6] For example on in which there are nine chips labeled with a different element element x, such that $x \in \{1, 2, 4, 8, 16, 32, 64, 128, 256\}$, and where the δ combination for this case occurs when an element d, such that $d \in \{7, 56, 73, 84, 146, 273, 292, 448\}$, can be expressed as the sum of some of the numbers selected by a player.

[7] This understanding derives from the normative characterization presented by Adrian Cussins in several conferences and seminars, with slight differences rooted in my understanding of similar remarks presented by John Haugeland (6).

[8] Heavy programming skills are not needed to design a simple program that can perform like a master in TTT. A fast google-search for TTT

playing TTT requires cognitive abilities to play, then, according to the assumption of cognitive science, such program must have some cognitive abilities, namely those that master human TTT players also have. But a TTT-playing computer can be excellent at playing simple TTT, and collapse in an TTT-variation[9]; and thus fail to pass the sphex-test. That is, a TTT-playing computer is prone to fail the sphex-test.

Why does the TTT-playing computer fail the sphex-test? It is because (a) the specification of the TTT domain is necessary in the design of an artificial agent and (b) such specification itself is enough to prescribe one unique representation and determine how appropriate behavior in the domain should be. If a and b hold, then (c) what is considered appropriate behavior for the program is completely fixed. But if c, a small change in the environment (such that produces a change in the valuation) could produce a massive collapse (*v.g.*, in the case that what was once incorrect behavior is now proper). Just like years of natural selection have determined the adequate behavior for the digger wasp, the designers determine what the adequate behavior of the TTT-program should be. And in the same way, the TTT-program is subject to the problems of the wasp, *i.e.*, fail the sphex-test.

It is obvious that an artificial agent cannot start completely out from zero. The designer has to determine *some* of its characteristics. Since the least amount of information needed to start to play TTT are the rules, the domain must already be fixed in the programming of the artificial agent. This means that thesis a cannot be rejected.[10] Then, what makes a TTT-playing program fail the sphex-test is that the domain of TTT itself imposes the constitutive and governant normativities (thesis b). That is, the determination of the domain (the rules of TTT) dictates what the elements of such domain are (the representation) and a unique method to achieve the goal. Accordingly we can say, in general, that what produces inflexibility in a given domain is that the constitutive and/or governant normativities get fixed by the domain. I will call this a TICTACness scenario: a case when the bounding normativity fixes the constituent normativity and/or the governant normativity. In the case of TTT the domain itself limits the representations to an equivalent set, and also how to produce δ conditions (how to win),*i.e.*, they determine a unique procedure to cope with the elements in a way that the goal is achieved.

Reconsider the case of Herbert and the sphex-test scenario formulated for it: a modification of the weight in the cans produces a massive collapse. A change in the weight of the cans produced a collapse of the appearance of cognitive abilities —Herbert completely ignores the goal and fails the sphex-test. The formulation of the sphex-test makes use of a small modification in the domain that cannot be accounted by Herbert's sensory apparatus (the representation of the domain[11]). Herbert can perform adequately in its domain because its inputs are associated with movements in a way that results in the fullfiling of its goal. Brooks' crew tweaks these associations in a way to guarantee it. But if there is an aspect of the domain for which there is no associated response, Herbert will not be able to deal with it. Precisely this is what happens in the sphex-test scenario: a new kind of input (the new empty can weight) for which the old procedures are not linked. This shows that the representations of the domain (Herbert's input-output associations) directly determine the possibility of success in the domain. In other words, Herbert exemplifies a case of TICTACness. There is only one way the domain is presented, thus being equivalent to the domain: whatever the representation cannot account is excluded from the domain. This leaves open the possibility for a sphex-test failure: if there is a change within the domain there is no way that Herbert can notice it, behaving exactly as if nothing had changed; with the result of ignoring the empty soda cans and thereby failing the sphex-test.

algorithms presents tons of possible solutions that can be easily reproduced.

9 Consider a tic-tac-toe game where the goal is to make the opponent make three in a line, or any other novel standards for achieving victory.

10 One could think that through methods like genetic algorithms a specification might be achieved that is not explicitly made by the programmer. Yet the method itself is determined (how a genetic algorithm works), so in a way the results are also determined, just in a broader way.

11 The representation of the domain is by no means a symbolic representation. The representation Herbert has is just the way that the world is presented to it: a bundle of information received through its receptors.

3 Achieving flexibility

We can now reexamine the intuitive formulation of the sphex-test we started from, making use of the characterization in terms of its normative constraints, in a way such that a possible solution begins to be visible. This will help us give a better explanation of what a small change is. A change, in general, is a modification of the initial situation of the artificial agent. And since the initial situation is characterized in terms of its normativities, a change in the situation is then a change with respect to the bounding, the constitutive or the governing normativity. Accordingly, a small change is one that (a) is not a change in the domain (a change in a domain would be a major change), thus falling within the scope of the bounding normativity and yet (b) does not comply to the constitutive and governing normativities. But if b, then such element is prone not to produce proper performance since the governing normativity cannot give a value to its contribution in reaching the goal. Thus, we can have a new formulation of the sphex-test:

> ***Sphex-test revisited***: If there is an element that falls within the bounding normativity for which there is not an element within the constitutive normativity that conforms with the governing, the alleged cognitive abilities correctly are only read into the agent.

This formulation presents a possible way out: in order to achieve the necessary flexibility to pass the test the presentation mode of the domain built in the agent must not have a predetermined connection with the steps necessary to achieve the goal the achieving of which will determine the grounds for cognitive abilities ascription. Such connection has to be created by the agent based upon some other mode of presentation that has a different goal and which has no evident relation with the first goal. This hypothesis tries to capture the fact, showed around the presented example, that cognition is involved both in the the act of *dealing* with a representation and *achieving* a certain representation. This is a fact ignored by several approaches in cognitive science: projects inscribed in the so-called good old fashioned artificial intelligence (SHRLDU(10), CYC(7), *Deep Blue*), others that make use of allegedly deictic representational systems (Pengi (1)), subsumption architectures (Herbert) or neural networks (NETtalk (8)), to name just a few. There is a failure to note the fact that the very same act of having a particular representation involves cognition, precisely the kind of cognition marvelously present in the artificial agents. There has to be a difference between the way the domain is presented and the way the domain is, such that being excluded from the representation does not necessary imply an exclusion from the domain. In other words, there must be space for error and change in the mode of presentation of the domain.

3.1 Possible ways to find a solution

One would imagine that stating the possibility of different goals within one same domain would be enough to dissolve the possibility of TICTACness. Let us examine two particular cases of a bounding normativity where two different governing normativities can apply.[12] One example of this can be seen in the relation between TTT and TTT^{-1}. TTT^{-1}(inverse TTT) is exactly like TTT except that the δ conditions for the players switch between them: one player wins when the *other* makes three in a line. This means that the domain for both games is exactly the same while their goals change. But as we saw above, the bounding normativity of the TTT domain fixes a unique set of equivalent representations; thus, if TTT^{-1} and TTT have the same domain, their constitutive normativities are also the same. A different example can be seen between *Atari-go* and regular *Go*. *Go* and *Atari-go* are oriental games played by two players who place black or white stones, respectively, anywhere on the intersections of a 19×19 grid, as long as the stone placed doe not recreates a former board state. Stones remain in the position they were placed unless they get surrounded by enemy stones, thus captured and consequently removed from the board. The goal of *Go* is to control the largest portion of the board (territory), while in *Atari-go* the goal is to capture the largest amount of enemy stones (regardless of the amount of territory controlled at the end of the game).

In the case of TTT and TTT^{-1} the goals, although different, are mutually inter-definable: one can understand TTT^{-1}'s goal in terms of TTT's goal. But this is not the case for *Atari-go* and *Go*: having the largest amount of territory does not necessarily amount to having captured the largest amount of stones and, likewise, capturing the largest amount of stones does not guarantee having the largest amount of territory. Since we want to avoid the cases where the representation alone assures compliance with the governing normativity, it is important that domains that can only have one set of mutually defineable goals are not selected. So we can say that a good sign that a domain is TICTACness is the impossibility to state different non-interdinable goals for which cognitive abilities that require cognitive abilities for proper performance. Yet having the possibility of having different goals does not guarantee flexibility, at least not on its own.

The sphex-test scenario presented shows how the possibility of different presentation modes of the domain is completely ignored, thus spoiling the appearance of cognitive abilities. Such failures are produced because the kind of performance expected is directly determined by the built in representation, product of their TICTACness. For instance, the behavior of the digger wasp in its original environment is the best course of action, but the inclusion of a small change (the movement of the prey) makes a change within the domain for which there is no adequate representation, such that what was the best course of action —what gave the appearance of intelligence— shows itself as absurd repetition (reveals the folly of the agent).

Maybe for natural agent, as the wasp, it is not easy to distinguish what falls into its domain and what does not. The evolutionary history of the organism and its niche may be not fully understood and so provide no easy way to draw the fine line between what is within its domain and what is not. However experimental setups are, or at least attempt to be, controlled setups where the task can be effectively done. Even in the case of Herbert where the domain are the MIT labs (not a simple fake setup like SHRLDU's, claims (3)) there are aspects excluded from the possibilities within the domain (for example, rivers insteas of corridors). The *quid* of the sphex-test is to present an example of a small change within the domain for which the agent has no adequate response but should have one. This means that in order to answer the challenge of the sphex-test it is necessary for the agent to have a way of dealing with the domain and to cope with unexpected elements (elements that are not initially represented). That is, that whatever presentation mode of the domain programmed (the representation the agent starts from) it must be possible to have other non-equivalent representations for that same domain. The sphex-test scenario are cases of elements that fall within the domain but do not fall within the representation the agent has of the domain. Thus, the way to solve this problem is for the agent to be able to handle, within the domain, different complete non-equivalent nor translateable representations. Such presentation modes of the domain have to be such that they are still presenting the *same* domain yet in a different manner.[13]

I will try to explain myself through yet another example. Consider drawing. The goal of this form of art, one could say, is to produce drawings, certain figures that resemble objects in the world. This simple form of art has clear definite bounding normativity that excludes other activities related with paper and pencil (*v.g.*, writting). Yet the governing normativity is not yet specified: nothing is said about the kind of figures and the methods that help achieve them. It is important to distinguish these two normativities: one determines the boundaries of the domain (what makes it drawing and not writting) while the other determines the way the domain is presented. Now take pencil and paper and draw a duck. The statement of this chore will only tell you the starting point (a blank sheet of paper) and end conditions (sheet of paper with a drawn duck). *How* to transform the blank paper into a drawing of a duck is yet to be determined: there is no fixed method to relate the final outcome to a sequence of pencil strokes starting from the empty paper. Having a paper and pencil and the goal of drawing a duck places you within the domain of drawing (excluding others like soccer, writting or reading). Consider now that you have a connect-the-dots diagram for the duck: it might require some skill but there is a clear method to relate the elements to the end (connecting with a line consecutive numbered dots). The dots determine which pencil strokes are correct and which ones not: they consitute one unique path to transform an empty sheet of paper into a drawing of a duck. A person may be very skillful in drawing a particular figure (following a diagram) and completely fail to build another figure (for which she has no diagram). If the only grounds for attributing cognitive abilities to that person was her ability to draw the duck, it would fail the sphex-test, leaving the ascription without grounds. This means that whatever elements an artificial agent starts from they must not be such that the main goal can be directly understood through them. Otherwise the cognitive workload is carried by the predetermined representation, and thus by the designer that predetermined such representation.

4 Conclusions

The situation of artificial agents is like Calvin's one in the comic strip from where the quote at the beginning in the introduction. In the strip, Calvin is complaining to Hobbes that he was forced to draw a duck by the connect-the-dots diagram. The point is that if such diagram is followed blindly, the result is fixed and thus inflexible, producing the

[12] It might be the case that the governing normativities be of the same kind (the same kind of guiding) but having different goals is enough to distinguish them.

[13] In order to continue the path presented in this work it is needed a full account of how is it possible to have different complete presentation modes of the same domain with different semantic values. Spelling out the full implications of this approach is out of the scope of this paper.

duck Calvin despises. Likewise, if the presentation mode of the domain is followed blindly (it is the only guide), the result is fixed and inflexible: TICTACness. The solution for both cases is to be able to use the initial presentation (the dots in Calvin's case) and construct something not directly specified: a novel drawing (product of connecting the dots in his own way) or a novel way of performing. Such novel constructions, ones that are not hidden within the original representation but constructed from them, are the ones that exhibit true cognition.

The sphex-test attempts to show that flexibility is needed for proper attribution of cognitive abilities. This flexibility is not a matter of adaptability, but of the normative structure of the experimental setup: the domain, the performance and the way the performance is achieved are crucial in determining if a given artificial cognitive agent is flexible or not. Accordingly, to succeed in the search of artificial intelligence it is not so important the complexity of the behavior but of the qualities of such behavior. For example, *Deep Blue*'s behavior can be highly complex but utterly useless in broadening our understanding of what intelligence is. If an artificial agent is to advance our understanding of cognition, it is necessary that it performs in a special domain, in a particular manner and with respect to a certain task. Thus, it seem that only if an artificial agent can pass the sphex-test can we hope to achieve artificial intelligence.

ACKNOWLEDGEMENTS

This work has been made within the *Perception, Coordination and Communication* research group of the Philosophy Department at the National University of Colombia. I would like to thank its members for fruitfull conversations around the subject. I would also like to thank Adrian Cussins, Carlos Márquez, Diana Acosta, Felipe Cuervo and an anonymous referee for their valuable comments on previous versions of this paper.

References

[1] Philip E. Agre and David Chapman, 'Pengi: an implementation of a theory of activity', in *Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI-87)*, pp. 268–272, (1987).

[2] Rodney A. Brooks, 'A robust layered control system for a mobile robot', *IEEE Journal of Robotics and Automation*, **2**(1), 14–23, (1986).

[3] Rodney A. Brooks, 'How to build complete creatures rather than isolated cognitive simulators', in *Architectures for Intelligence*, pp. 225–239. Erlbaum, (1991).

[4] Rodney A. Brooks, 'Intelligence without representation', *Artificial Intelligence*, **47**, 139–159, (1991).

[5] Adrian Cussins, 'Why a closed, rule-governed, digital system need not be a formal system: an argument concerning the ancient game of go, the nature of computation and the distinction between intelligent creatures and mere brutes', (1999).

[6] John Haugeland, *Having Thought. Essays in the Metaphysics of Mind*, Harvard University Press, Cambridge, Massachusetts, 1998.

[7] Douglas B. Lenat and R. V. Guha, *Building Large Knowledge Based Systems: Representation and Inference in the Cyc Project*, Addison-Wesley, 1990.

[8] T. J. Sejnowski and C. R. Rosenberg, 'Parallel networks that learn to pronounce english text', *Journal of Complex Systems*, **1(1)**, 145–168, (1987).

[9] Claude E. Shannon, 'Programming a computer for playing chess', 2–13, (1988).

[10] Terry Winograd, 'A procedural model of language understanding', in *Computer Models of Thought and Language*, eds., Roger Shank and Kenneth Colby, W. H, Freeman, (1973).

[11] Dean Woodridge, *The Machinery of the Brain*, McGraw Hill, Ney York, 1963.