Journal of Artificial Intelligence and Consciousness

A Relational-Database Methodology for Incorporating Consciousness into AI Agents

Ouri Wolfson

Department of Computer Science, University of Illinois at Chicago, 851 S. Morgan St.
Chicago, IL 60607, USA
wolfson@uic.edu

Consciousness is key to Artificial General Intelligence. It is currently unclear what is consciousness and we expect that by the time it is understood, there will probably be millions of man-years invested in the development of AI agents. This paper addresses the following question. How should AI agents be engineered from now on, such that when it becomes clear how consciousness and subjective experiences are generated, the effort of endowing AI agents and robots existing at that time with consciousness is minimized. As far as we know this question has not been studied heretofore. The paper proposes a methodology and architecture that are based on evolving Probabilistic Relational Databases, and on stepwise progressive data fusion. The methodology and architecture are independent of what consciousness turns out to be, provided that it can be incorporated into AI agents.

Keywords: Probabilistic Relational Databases;

1. Introduction

Broadly speaking, the moment to moment subjective experiences (e.g. the feeling of sadness, the smell of coffee, the sound of a song, the sight of the color red) of a person aggregate to form his/her consciousness. Philosophers and neuroscientists starting with Descartes ("I think therefore I am") agree that it is providing the feeling of life and some argue that consciousness is the only thing that matters personally [1]. However, what is the functional purpose of an experience? Roger Penrose, among others, argues that human level General Artificial Intelligence is not possible without consciousness [2]. So does Antonio Damasio: "I suggest only that certain aspects of the process of emotion and feeling are indispensable for rationality" [3, p. xiii], Wah and Chi [22], and Reggia et. al. [26]. We agree and we postulate that moral and ethical decisions also require consciousness. For example, consider a nursing home patient being nursed by a robot. We argue that the care will be more effective if the patient knows that the robot has feelings, i.e. is conscious. While an unconscious robot may help the patient to get up after a fall, only a conscious one may credibly also say: "I can understand your pain".

Unfortunately, it is currently unclear what consciousness is. However, some progress is made by neuroscientists, roboticists, and philosophers. For example, two currently leading theories are the Workspace Model [4, 20] and Integrated Information Theory (IIT) [25,1,5][a]. IIT postulates that today's computers are not conscious, however, consciousness may emerge from other computational paradigms, e.g. information processing by spiking

[a] A competition is currently underway to determine the more appropriate between the two [7].

neural networks [1]. Other classes of theories of consciousness that are implementable in AI agents are Internal self-models, Higher-level representations, and Attention (see [8]).

In this paper we ask the following question: assuming that some mechanism is proven to provide consciousness, how should this knowledge be used in order to endow an existing AI agent (e.g. a robot) with consciousness? More specifically, consider the following postulate: by the time artificial conscious functions are built, there will probably be millions of man-years invested in the development of AI agents. Furthermore, we postulate that without proper planning, incorporating the discovered consciousness mechanism into an existing AI agent will be a monumental task, requiring the reengineering of the entire agent. Therefore, this paper addresses the question: how should existing AI agents be engineered now, such the reengineering effort expended to endow them with consciousness when the phenomenon is understood is minimized? Towards this goal, the paper proposes building blocks and an AI architecture that will enable incorporation of consciousness without reengineering of the entire agent; the architecture will enable the incorporation of consciousness by building a relational data integration module[b]. In other words, if the architecture is obeyed by current AI builders, then incorporation of consciousness later on will amount to writing a data integration module, which should be simpler and less time consuming than reengineering the entire agent. Therefore, this paper should be regarded as an engineering proposal, in the spirit advocated by Holland [24].

More specifically, our architecture models the computational output of an unconscious agent A by a probabilistic relational database, and assumes that consciousness is provided by some oracle that is invoked by A. We remain agnostic to the nature of the oracle and treat it as a black box[c]. In addition to providing subjective experiences, the oracle O also produces a probabilistic relational database called the materialized intuition (*mi*). The database explicates a subjective experience by digitizing the metadata of subjective experiences, e.g. the quale red will be represented by the string ‘red’ in the database. And feelings, intuitions, etc., will be similarly digitized. Then the incorporation of consciousness into the agent A amounts to the following: 1. Including the oracle O in the agent A, and 2. Integration of the databases produced by the unconscious agent A and that of the oracle.

The feasibility of producing the *mi* database is supported by the fact that humans can recognize that they are conscious and having subjective experiences, and that they can describe and encode these experiences. It is true that animals are also probably conscious,

[b] This type of integration is precisely defined in sec. 2.3 and is different that the information integration of IIT. More specifically, IIT proposes that consciousness is generated by some form of information integration. Although interesting, this hypothesis has not been proven and is still controversial. In contrast, we propose a mechanism by which Artificial Consciousness can be embedded in existing AI agents and robots, regardless of how consciousness is generated.

[c] In fact, existence of a consciousness oracle was hypothesized by Holland (called consciousness module [24], p.85). Thus the present paper can be regarded as making the idea concrete, including the relationship of the oracle to other architectural components of the agent, and the associated data structures and communication mechanisms.

but they cannot encode experiences. However, remember that the Oracle O is human-engineered. And although the generation of the subjective experience itself probably cannot be programmed into existing computers (although it may emerge from programming in some architectures), we postulate that the recognition of subjective experiences by the agent having them can be programmed; and furthermore, we postulate that the coding of such experiences as prd tuples can also be programmed. In fact, this postulate, which we call "***Recognize and Code"*** was articulated previously by Haikonen (see [9] p. 196):

"The test for machine consciousness would now be that (a) the machine is able to report that it has inner imagery and inner speech and (b) it can describe the contents of these and recognize these as its own product."

Similarly, Aleksander indicates [17]:

"**The display characteristic**. *A machine in the CM [Conscious Machines] category has a designer/user who, to indicate that the machine is conscious, includes, as part of the design, means for communicating outwards what the machine is conscious of at a point in time in terms of displaying its internal state*."

In our formulation the <u>means for communication</u> is a probabilistic relational database (mi). One may argue that an unconscious agent can also pretend to be conscious and produce the materialized intuition. However, we assume that the oracle O is not preprogrammed to lie about having a subjective experience. This assumption, which we call ***"Honesty",*** was also made previously (see [9] p. 196):

"It would also have to be known that the machine does have concepts like 'I', 'to have' and 'inner imagery'. Thus here the mere reproduction of blindly learned strings of words like 'I have inner imagery' would not count as a proof."

Furthermore, recall that the purpose of this paper is not to propose a mechanism for producing artificial consciousness, or to prove one as such; it is to propose a consciousness-incorporation mechanism that is applicable <u>after</u> artificial consciousness has been proven.

There has been prior work on the architecture of conscious robots (see [10,11]). However, those works make certain assumptions on what consciousness is, and propose an architecture that will produce it. However, at this time, artificial consciousness is very controversial and there are multiple candidate approaches [8]. Our proposed architecture is applicable regardless of the approach that will produce artificial consciousness. In our vision, the methodology of incorporating consciousness into AI agents (existing or future built) amounts to a data encoding and integration problem, whatever consciousness eventually turns out to be.

The rest of this paper proposes the Model in Sec. 2, discusses computations and consciousness oracles in sec. 3, and concludes in Sec. 4.

2. The model

This section discusses the building blocks of the proposed methodology. Specifically, in sec. 2.1 we define Probabilistic Relational Databases, and in sec. 2.2 we discuss the usage of these databases and the main concepts of the methodology.

2.1 Probabilistic relational databases

Our model uses relational databases to represent knowledge and activities of conscious and unconscious entities, and associated probabilities to represent the fact that some of the knowledge may be uncertain. A *probabilistic relational database* (prd) is a relational database in which each tuple is associated with a probability [12, 13]. The probability denotes the confidence in the fact represented by the tuple. For example, the tuple *likes(jane, joe, 0.3)* indicates that Jane likes Joe with a probability of 0.3. A probabilistic relational database satisfies the following constraint:

Constraint 1: Consider a probabilistic relation *pR*, and the deterministic relation *R* that is obtained after projecting out the probability attribute. If *S* is the set of tuples in *pR* that have the same *R*-key (i.e. key of the relation *R*), then the probabilities of all the tuples in *S* sum up to at most 1.

For example, if the tuples with key 'Joe' in the relation salary are (joe, 51K-60K, x)[d] and (joe, 61k-70K, y), then $x+y \leq 1$.

Furthermore, a tuple can have a single probability, i.e., the relation likes cannot have 2 tuples likes(jane, joe, 0.2) and likes(jane, joe, 0.3). This is reflected by the next constraint:

Constraint 2: If $a_1,\dots,a_n$ are attribute values of a relation *r*, and *x, y* are probabilities where $x \neq y$, then r cannot have two tuples r($a_1,\dots,a_n$,*x*) and r($a_1,\dots,a_n$,*y*).

Observe that a deterministic relational database is a prd in which the probability of each tuple is 1. We assume the closed world assumption. This means that if likes(jane, joe, 0.3) is a database tuple, then there are no other tuples of the form likes(jane, joe, p) in the database, and with probability 0.7 Jane does not like Joe.

2.2 Entities and world models

There is a set of processing and communicating units called *entities*. Intuitively, humans, animals, and computers are examples of entities. Time is divided into intervals (or time units) t1, t2,… In a time interval an entity receives an input, performs computation and communication, and produce an output. Conscious entities (defined below) also have experiences in each time interval.

The input of each entity at a time interval is a probabilistic relational database consisting of data received from two sources:

1. communication from other entities (see the transmission step sec. 3.3), and
2. the entity's senses.

[d] We use two notations for a tuple. When the attribute names are self-evident we omit the attribute names (e.g. *likes(jane, joe, 0.3)*), otherwise we make them part of the tuple (e.g. *likes(Subject=jane, Object=joe, Probability = 0.3)*)

For example, some tuples in the input prd may be:

Patient(CurrentLocation=kitchen, prob=0.1); this tuple means that the patient is in the kitchen with probability 0.1.

Patient(CurrentLocation=den, prob=0.9); this tuple means that the patient is in the kitchen with probability 0.1.

Honda(direction=NE, Distance=200m, prob.=1.0); this tuple means that there is a car of type Honda which is 200 meters in the North-East direction.

Observe that some information normally associated with subjective experiences may be part of the input prd. The reason for this is that meta-data of sensory input, e.g. Color(Red), can be recognized by computable functions and can be part of the input even if the entity is unconscious (e.g. a computer). This subset of the input is called the *experiential component* of the input o, and is denoted *ec(o)*.

As indicated, during a time interval an entity receives an input, performs computation and communication, and produces an output. The computation is further divided into the Input Processing step and the Integration step, both of which are discussed in sec. 3. By communication we mean receiving an input from other entities at the beginning of the time interval, and sending to other entities part of the output at the end of the time interval (see fig. 1).

Over an entity's *e* history, the computation and integration steps of *e* build a world model that represents *e*'s knowledge. In other words, over the time-intervals, the input, computations and integrations of *e* produce *e*'s world model denoted *e(wm).* A *world model* is a probabilistic relational database. Intuitively, *e*'s world model at a time is a summary of its prior inputs, computations, and experiences.

Some tuples in the world model are actuators, in the sense that they produce an action in the real world, e.g. if the tuple open(door, kitchen) is inserted in the prd, the result is the activation of an actuator that opens the kitchen door; similarly there is a tuple that results in the robot kicking a ball, etc.

The world model (which is empty at time 0) is modified by the computation and integration steps of each time interval. Therefore, *e(wm)* at a point in time represents the knowledge and actions of *e* in the same way that the database of a bank represents its knowledge and actions at that time; Or, the way an airline database represents the world knowledge of its customers, employees, flights, cancellations, investors, assets, etc.

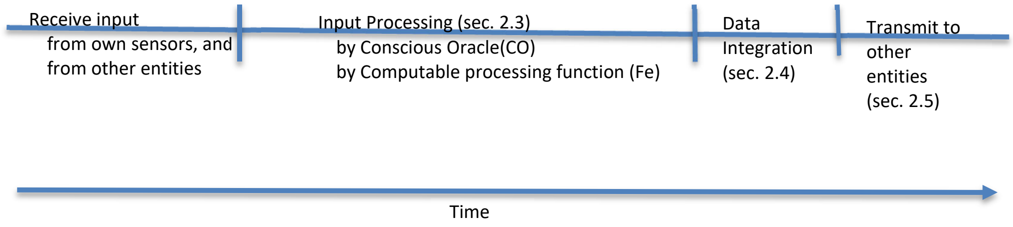

Figure 1: An entity's sequence of activities during a time interval.

This model is consistent with existing consciousness theories. For example, Verschure proposes that "Given the above analysis we can define consciousness as a transient autonomous memory, which maintains a delayed, virtualized, unitary representation of the agent–environment nexus. This representation can be decomposed along the three columns and layers of the DAC architecture."[V2016, p.9]. We postulate that this memory can be represented as a prd.

We assume that the computation and integration steps produce consistent world models. There are two types of world-model consistency, syntactic and semantic. Syntactic consistency means that the wm does not contain conflicting tuples, e.g. Patient(CurrentLocation=kitchen, prob=1), Patient(CurrentLocation=den, prob=1), which indicate that the patient is in both the den and the kitchen. Syntactic consistency can be automatically enforced through constraints on the prd, such as those defined in sec. 2.1 and traditional database constraint mechanisms such as key constraints[e].

Semantic consistency means that it does not represent real-world contradictions. Intuitively, a world model is semantically consistent if a person's finger is in a fire and the person is in a subjective state of pain, i.e., it contains the tuples Finger(burning), Pain(10); the world model of the person is semantically inconsistent if the person's finger is in a fire, the person is in pain, and her mood is serene; i.e. it contains the tuples Finger(burning), Pain(10), Mood(serene). Semantic consistency is not enforced, although some level of conflict resolution can be engineered into the data integration step. We discuss this further in sec. 3.2.

3. Computations and Oracles

In this section we discuss the three steps that sequentially comprise the computations at each time unit: Input processing (sec. 3.1), the information integration (sec. 3.2), and the transmission (sec. 3.3). The first step also has a non-computational component represented by the consciousness oracle.

3.1 The Input Processing step

The *Input Processing step* of the time-interval invokes two engines: a computable processing function (the intelligence engine), and a consciousness oracle. Each entity e has a *computable processing function,* or a *processing function* for short, *Fe*, at each time interval. The processing function at time *t*, Fe_t, maps its input *o* and world model at time *t* into another world model at time *t+1*. Furthermore, the processing function at time *t+1* may be different than the processing function at time *t* due, for example, to the change of some weights in a Neural Network. Thus, $Fe_t(o,wm)=(wm', Fe_{t+1})$.

[e] Which means that the database should not accept two different tuples with the same key.

In addition to the processing function, conscious entities also have a *Consciousness Oracle (CO)*. Intuitively, the consciousness oracle of entity *e* takes *e*'s input prd and its world model, and it produces: 1. a subjective experience, and 2. a materialized intuition; the latter is a prd that captures the result of the experience. See fig. 2.

We treat the *CO* as a black box. It is possible that the *CO* has its own internal database that is not exposed to the other architectural components of the entity, and may not even be digital. The *CO* also encapsulates what we currently call the subconscious processing of conscious entities.

In this paper we are not concerned with the question: how is the subjective experience produced. However, our model is consistent with the materialist or dualist perspective in the sense that the *CO* may either: 1. produce an experience itself, or 2. allow the entity access to some pre-existing experience in a "global consciousness pool", or 3. interact with some unknown modules that produce subjective experiences. Furthermore, our model is consistent with richer interpretations of consciousness, for example one which produces, in addition to subjective experiences, "understanding" in the Penrose sense [2].

In the case of unconscious entities (e.g. nowadays robots and other AI agents) the conscious oracle is empty. More precisely, a *conscious entity* is one which has a consciousness oracle, whereas an *unconscious entity* is one which does not do so.

Formally, an *experience,* also called a *subjective state*, is a structured set of *qualia*, where a quale is, for example, a sense of self, a color sensation, a sound sensation, a smell sensation, etc. [1]. Each quale *q* has associated with it *meta-information* denoted *m(q)*, e.g., pain level is 3, or m(q)='red'. The *meta-information* of an experience *s*, denoted *m(s),* is a set of tuples, each describing a quale of *s*. For example, Pain(3), Color(Red), Color(Blue), Smell(coffee,2); these indicate that the pain level is 3, colors experienced are red and blue, and the smell of coffee has intensity 2. Thus, *m(s)* is also a prd. This prd may also represent the structure of the qualia, such as a graph. For example, the tuple (Pain(3), Color(red)) represents an edge between the Pain vertex and the Color vertex in the graph labeled 'wound'. Obviously, since an experience *s* is a set of qualia, *m(s)* is not part of *s*. However, *m(s)* is part of a separate database called the materialized intuition prd, which is precisely defined below.

Formally*, the Consciousness Oracle* of entity *e* at time *t,* denoted COe_t, is a function $COe_t(o, wm)=(s, mi, COe_{t+1})$ where: *o* and *wm* are the input and the world model of *e* at the beginning time interval *t*; *s* is the subjective state of *e* at time *t*, *mi* is the materialized intuition (defined below), and COe_{t+1} is the conscious oracle of e in the next time unit. Observe that, in the same way that the processing function at time *t+1* may be different than the processing function at time *t,* we allow for the possibility that COe_{t+1} is different than COe_t. Although the question of how the oracle is modified from one time unit to the next is beyond the scope of this paper, we make the following related observation.

There are indications that consciousness arises from some coordination in transition between brain states [14, 15]. For example, in [15] we have shown that traffic in the human brain seems to be globally coordinated. This conclusion was derived from the fact that, unexpectedly, traffic in the brain is closer to a System Optimum than to a User Equilibrium (a game theoretic notion generally accepted to model traffic in the absence of some global coordination)[f]. However, the source of the global coordination is unclear. If coordination of traffic in the brain is linked to consciousness, this will indicate that consciousness derives from transition between states, rather than particular states of a system such as the brain. We hypothesize that traffic coordination is linked to consciousness, and intend to test this by repeating the traffic analysis for unconscious patients, as well as progressively more primitive life forms such as animals. Regardless of the result of the test, the model of this paper supports the link between state transitions and consciousness in the sense that the subjective state may arise due to transition from COe_t to $COe_{t+1.}$ Furthermore, this view as well as our model is consistent with consciousness being a "spatiotemporal pattern in a specific physical substrate" [21]. In particular, this would simply mean that the Consciousness Oracle is implemented in that substrate. For example, consider IIT, and assume that, as IIT postulates, consciousness is indeed "the way information feels when it is processed a certain way", i.e. a way that satisfies the IIT axioms, e.g. by a spiking neural network [1]. Then the CO will be a spiking neural network that takes as input the pair *(o, wm)* and produces the subjective experience *s*, and the digital output *mi*.

Another observation is that Fe_t may modify *wm* to produce *wm',* but the input to COe_t is *wm*. In other words, if Fe_t modifies *wm*, then all the modifications are applied after both Fe_t and COe_t have completed in the time interval *t*. Furthermore, the order of Fe_t and COe_t invocations during the Input Processing step is immaterial in our model; i.e. they may work sequentially or in parallel. Or, Fe_t may invoke COe_t, or vice-versa.

The *materialized intuition (mi)* is a prd that contains the metadata of *s*, namely *m(s)*. Additionally, *mi* may contain tuples that capture intuitive information, such as:

(1) "with probability 0.8 the patient has the flu"; or

(2) insert the tuple verbalize("I can understand your pain") (an actuator tuple).

It is important to note that it is possible that the *CO* performs unknown "computations" and produces *mi* tuples that are initially meaningless and unintelligible to the outside world. For example, we assume that "it feels like something" to be a bat, but humans cannot understand this feeling. Therefore, if the *CO* produces similar feelings, and these feelings are encapsulated in the *mi*, it is possible that initially they are meaningless to Fe_t or to humans that programmed it. Such an unintelligible *mi* tuple is called a *u-tuple*.

Although initially u-tuples are meaningless, over time their meaning may be understood via machine learning, pattern matching, and reverse engineering by the entity's (e.g. AI agent's) human builders and owners. A simple example of a u-tuple would be one generated by a CO that is able to perceive, i.e. obtain a subjective experience, from external sources that are not perceivable by humans, such as infrared electromagnetic waves, or

[f] Notice that the notion of coordination used in [15] is different than the traditional one in the following sense. In existing literature, coordination and correlation in the brain refers to different brain regions being active concurrently, or according to some spatio-temporal pattern. In contrast, [15] refers to coordination of signal traffic, i.e. the signals that travel along the structural connectome to activate/deactivate brain regions. And this signal-movement turns out to be globally coordinated for the purpose of traffic-efficiency.

ultrasound. In this case, the *CO* may generate as part of the metadata *m(s)*, a prd tuple such as *color(xx)*; *xx* is a code indicating that natural language does not have a word for the perceived color. However, over time it can be learned that *xx* is associated with a particular infrared electromagnetic wave-length. Of course, doing so for higher level emotions will be more difficult but not impossible.

Such reverse engineering may eventually provide an interpretation of "what it is like to be a bat" [16] in a sense hypothesized by Holland [24, p.87]: "…any subjective view must be represented, and anything that is represented can in principle be known - in other words, we might be able to say 'what... it resembles' by giving a more or less exact description."

3.2 The Integration step

Data integration has been studied extensively in the data management community (see e.g. [18.19]). It is the process of integrating data from different sources, and it is summarized in [19] as follows:

"… an integration scenario with a three-step data integration process.... First, we need to identify corresponding attributes that are used to describe the information items in the source. The result of this step is a schema mapping, which is used to transform the data present in the sources into a common representation (renaming, restructuring). Second, the different objects that are described in the data sources need to be identified and aligned. In this way, using duplicate detection techniques, multiple, possibly inconsistent representations of the same real-world objects are found. Third, in a last step, the duplicate representations are combined and fused into a single representation while inconsistencies in the data are resolved."

Recall that in our vision the consciousness oracle and the processing function (the data sources in [19] terminology) are developed independently by different teams, and the developments are possibly separated by many years. Since, the outputs of COe_t and Fe_t may overlap, they need to be integrated according to the above process. For example, *m(s)* produced by the CO may overlap with the *experiential component* of the input *ec(o)* in the sense that both contain a Color tuple; the former as a result of the subjective visual experience, and the latter as a result of the electromagnetic spectral analysis. Or, Fe_t may compute the location of the patient, her physical status (standing, walking, sitting, or fallen), and her mood; whereas COe_t may produce AI agent *e*'s intuition about the patient's mood. In this case, the mood tuples may conflict.

For another example, consider the experiential component of the input *o, ec(o)*. Assume that it is transferred by Fe_t to become part of *wm*; it may contain the tuple *kite(color=purple, prob=0.7),* whereas *m(s),* which is part of *mi*, may contain the tuple *kite(color=purple, prob=0.9)*. This is a conflict since if both tuples are incorporated into the resulting *wm'*, its prd representation would violate **constraint 2**. Intuitively, the conflict means that the entity believes that the kite is purple with two different probabilities. Or, the *wm* may contain the tuple *kite(color=<u>purple</u>, prob=1)* whereas *m(s)* may contain the tuple *kite(color=<u>red</u>, prob=1)*. Or, in the case of the nursing-home robot, the *mi* may contain the tuple *patient(in-pain, prob=1)* whereas the *wm* has the tuple *patient(in-pain,* *prob=0.1).* This conflict indicates that computationally the robot is sure that the patient is in pain, but its intuition is that the patient is in pain with only a small probability.

These conflicts are resolved during the integration step of a time interval. Intuitively, this step integrates the *mi* prd and the *wm* prd of entity *e* at time *t.* This step is executed by an *Integration Program (IP)* written specifically for each AI that is to be endowed with consciousness. The IP will handle conflicts between the *mi* and *wm*. A simple case is when the *mi* contains tuples that are not originally in *wm*, but need to be integrated into it. For example, the actuator tuple: (verbalize "I can understand your pain") may exist in the *mi* but not in the *wm*. Then the IP will probably transfer the tuple from the *mi* to the *wm* as is.

Now consider the case that *mi* contains the tuple *patient(in-pain, prob=1)* whereas the *wm* contains the tuple *client(in-pain, prob=0.1).* This means that based on the latest input the Consciousness Oracle is certain that the patient is in pain, whereas the world model of the entity believes that this is the case only with probability 0.1. Moreover, the CO calls the client "patient", but both the CO and Fe_t refer to the same individual. Here, the IP will have to first determine whether the two prd's refer to the same individual, and if so, make a decision as to the pain probability. This is a labor intensive programming effort, although less intensive than rewriting Fe_t and incorporating CO into it. Some straightforward approaches to data integration may be: 1. to believe the intuition and ignore the world model, or vice versa, i.e., 2. to believe the world model and ignore the intuition, or 3. some middle ground such as to make the pain probability 0.55. If the 3rd approach is taken, then the probability of the tuple in the world model is modified to *patient(in-pain, prob=0.55).*

Generally, the *mi* prd and the *wm* prd need to be integrated in a way that ensures that the *wm* of a conscious entity remains syntactically consistent at each time interval *t* (see sec. 2.2 for the definition of syntactic consistency). Semantic consistency is a target of the integration step, however it may not be fully achieved. And this is justified by the fact that conscious entities may be conflicted on some issues.

The proposed system architecture is illustrated in Fig. 2 below.

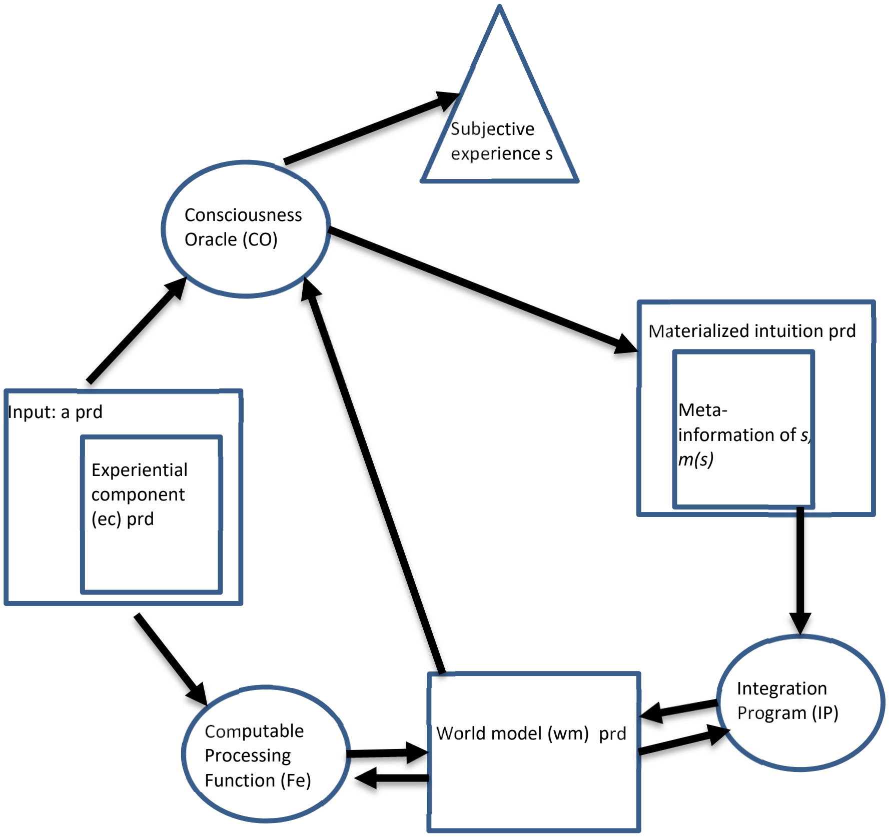
Subjective experience s
Consciousness Oracle (CO)
Materialized intuition prd
Meta-information of *s*, *m(s)*
Input: a prd
Experiential component (ec) prd
Integration Program (IP)
Computable Processing Function (Fe)
World model (wm) prd

Figure 2: System Architecture: Processes, their inputs and outputs

3.3 The Transmission step

After the computation step of a time interval completes, each entity performs the *transmission step* in which it transmits parts of its world model to other entities. The subset of entities to which an entity *e* transmits information during a time interval may be empty. The information transmitted by *e* to different entities may be distinct, i.e. it may transmit one tuple to entity e_1 and another tuple to entity e_2. These tuples are received by e_1 and e_2 at the beginning of the next time interval.

Recall, experiences are *not communicable*. This means that entities cannot transmit subjective states to each other, and cannot identify a subjective state to each other. In particular, an entity cannot transmit or identify its own subjective state (beyond the meta-information associated with the state) to another entity.

4. Conclusion and Discussion

Presently, there are multiple approaches to consciousness and there is no proof as to what constitutes consciousness, and how it is generated by conscious entities. However, we and many others postulate that consciousness is necessary for Artificial General Intelligence and that it can also improve Special-Purpose AI.

Given this state of affairs, our paper considers a future in which the consciousness problem has been solved in the sense that it has become clear how it is generated. We are agnostic as to the nature of the solution, it may involve dualism, materialism, or a combination of the two. Our model captures the generation of conscious experience by a black box called the Consciousness Oracle (CO). And the model positions the CO within a proposed architectural framework of an AI agent. The merit of the framework is that it minimizes the effort of building consciousness into AI agents and robots that are built according to the framework. Specifically, this effort will have two components:

(1) Replacing the CO black box by the consciousness generating mechanism that has been discovered, and

(2) Developing an Integration Program that is invoked at each time unit to merges two probabilistic relational databases (prd's).

Our proposed framework models an AI agent, i.e. its input, knowledge, output and actions by a set of probabilistic relational databases (prd's). The databases are probabilistic due to the fact that the agent's knowledge may be uncertain. An important postulate and motivation for the work is that the pre-existing AI engine of the entity *e*, denoted here Fe_t, will not need to change in order to incorporate consciousness.

To some, the idea that intelligence/function/computation can be separated from consciousness according to the model of this paper may seem naive. However, consider that on the other hand, the idea that if and when consciousness is understood, every intelligent agent existent at that time will need to be re-implemented in order to incorporate consciousness seems equally naive.

What is the relationship of our proposal to the Global Workspace (GW) model [23]? Is the collection of probabilistic relational databases the GW? Certainly the wm database contains global knowledge that goes beyond a workspace. However, at a time unit *t*, the combination of the input database, the materialized intuition (mi) database, and the output of the intelligence engine Fe_t can be viewed as the current workspace. And, although it may be true, our model does not claim, as GW, that this combination constitutes consciousness; in other words, usefulness of the model does not depend on such claim.

What would adoption of our proposal mean? It would mean that AI agents will be built according to the proposed architecture, in order to facilitate future endowment of the agent with consciousness. Importantly, we feel that conforming to the architecture does not require novel AI concepts or a particularly large effort. The reason is that the components in the framework (the prd's) must already exist in any AI design, although they may be represented by different data structures and modularized differently. Thus our proposed architecture simply standardizes data structures as evolving prd's and modularizes software into components such as input ingestion and processing, integration into existing knowledge, and communication. Furthermore, the proposed architecture will allow embedding of a Consciousness Oracle in existing robots for testing and experimentation with various consciousness generating approaches.

References

[1] C. Koch, "The feeling of life itself: Why Consciousness is Widespread but Can't be Computed", MIT press, 2019.

[2] R. Penrose, "Shadows of the mind: a search for the missing science of consciousness", 1994, Oxford University Press.

[3] A. Damasio "Emotion, Reason, and the Human Brain", Grosset/Putnam, 1994.

[4] S. Dehaene, S., C. Sergent, and J. Changeux, . A neuronal network model linking subjective reports and objective physiological data during conscious perception. Proc. National Academy of Science (USA) 100. 14: 8520-8525.

[5] http://integratedinformationtheory.org/

[6] P. Haikonen, "Robot brains: circuits and systems for conscious machines". John Wiley & Sons, 2019.

[7] P. Ball, "Neuroscience Readies for a Showdown Over Consciousness Ideas", Quanta Magazine, March 2019

[8] J. Reggia, "The rise of machine consciousness: Studying consciousness with computational models", Neural Networks 44, 2013, pp. 112–131

[9] P. Haikonen, Robot Brains : Circuits and Systems for Conscious Machines, John Wiley & Sons, Inc., 2007.

[10] Y. Kinouchi, K. Mackin, A Basic Architecture of an Autonomous Adaptive System With Conscious-Like Function for a Humanoid Robot. Front. Robot. AI 5:30. doi: 10.3389/frobt.2018.00030, 2018

[11] P. Verschure, Synthetic consciousness: the distributed adaptive control perspective. Phil. Trans. R. Soc. B 371: 20150448. http://dx.doi.org/10.1098/rstb.2015.0448, 2016.

[12] R Cavallo, M Pittarelli, The theory of probabilistic databases. Proc. of VLDB 1987.

[13] N. Dalvi, D. Suciu, Efficient query evaluation on probabilistic databases. VLDB Journal 16(4), 2007.

[14] A. Demertzi, E. Tagliazucchi, S. Dehaene, G. Deco, P. Barttfeld, F. Raimondo, C. Martial, D. Fernández-Espejo, B. Rohaut, H. U. Voss, N. D. Schiff, A. M. Owen, S. Laureys, L. Naccache, J. D. Sitt, Human consciousness is supported by dynamic complex patterns of brain signal coordination. Science Advances 5, eaat7603, 2019

[15] O. Wolfson, P. Szczurek, A. Vijayan, A. Leow, O. Ajilore, ”A Traffic Analysis Perspective on Communication in the Brain”, Proc. of the 18th IEEE International Conference on Mobile Data Management (MDM), Daejeon, South Korea, May 2017, pp. 206-211.

[16] T. Nagel, What is it like to be a bat. Philos. Rev. 83, 435–450. 1974 (doi:10.2307/2183914)

[17] I. Alexander, The category of Machines that Become Conscious. Journal of Artificial Intelligence and Consciousness Vol. 7, No. 1 (2020), World Scientific Publishing Company.

[18] NAUMANN, F., FREYTAG, J.-C., AND LESER, U. 2004. Completeness of integrated information sources. Inf. Syst. 29, 7, 583–615.

[19] J. BLEIHOLDER and F. NAUMANN, Data Fusion, ACM Computing Surveys, Vol. 41(1), Dec. 2008.

[20] B. J. Baars, A Cognitive Theory of Consciousness, Cambridge University Press, Cambridge, UK, 1988

[21] D. Gamez, The Relationships Between Intelligence and Consciousness in Natural and Artificial Systems, Journal of Artificial Intelligence and Consciousness Vol. 7, No. 1 (2020), World Scientific Publishing Company.

[22] N. G. Wah and L. W. Chi, Strong Artificial Intelligence and Consciousness, Journal of Artificial Intelligence and Consciousness Vol. 7, No. 1 (2020), World Scientific Publishing Company.

[23] B. J. Baars, Global workspace theory of consciousness: toward a cognitive neuroscience of human experience?, Progress in Brain Research, Vol. 150 , 2005 Elsevier

[24] O. Holland, Forget the Bat, Journal of Artificial Intelligence and Consciousness Vol. 7, No. 1 (2020), World Scientific Publishing Company.

[25] M. Oizumi, L. Albantakis, and G. Tononi, From the phenomenology to the mechanisms of consciousness: Integrated information theory 3.0, PLoS Computational Biology 10(5), 2014.

[26] J. Reggia, G. Katz, G. Davis, Artificial Conscious Intelligence, Journal of Artificial Intelligence and Consciousness Vol. 7, No. 1 (2020), World Scientific Publishing Company.