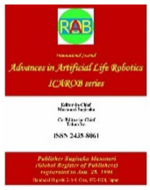
SUGISAKA MASANORI

Journal of Advances in Artificial Life Robotics
Vol. 2(3); December (2021), pp. 158-161
ON LINE ISSN 2435-8061; ISSN-L 2435-8061
https://alife-robotics.org/jallr.html



Research Article

Inter-Induce computation and its Philosophical Foundation

Yasuhiro Suzuki
Graduate School of Informatics, Nagoya University, Furocho Chikusa Nagoya City, Aichi Prefecture 464/0814, Japan

ARTICLE INFO

Article History

Received 31 December 2020
Accepted 05 March 2022

Keywords

Inter-induced computation
Natural computing
Foundation of computer science
Heart sutra

ABSTRACT

Set theory is based on the distinguishability of elements. How to recognize and identify the world is the essence of set theory. If each element cannot be identified, all the elements are one set. So the set does not make sense. The Heart Sutra is highly rational and can be interpreted mathematically. The mathematical interpretation of the Heart Sutra shows the divergence of how to discriminate. Based on this world view of Heart Sutra, we propose Inter-Induce computation, IIC as a novel calculation paradigm that does not depend on set theory. This paper gives an overview and philosophical foundation of IIC.

1. Introduction

There are several types of the calligraphy of a circle (Enso 円相 in Japanese), for example, Zen master Ikkyu painted a perfect circle (Fig.1 left), while Zen master Hakuin or Buddhist preast Sengai drowned not a perfect circle, but the circle is broken in one place (Fig.1 right).

We do not know the beginning and end of a perfect circle, but if the circle is broken in one place, we know the point of beginning or the end. And we know the beginning and end of a broken circle from brushstroke (Fig.1 right).

Topologically, a perfect circle with one point broken is a straight line. It creates "time" because a straight line has a beginning and an end. In other words, when a perfect circle is broken in one place, time is created. The perfect circle is a time-integrated dynamical system, and breaking the perfect circle causes the dynamical system to evolve in time. The phenomenological equations of the dynamical system are known, but the phenomenological equations of this world are unknown.

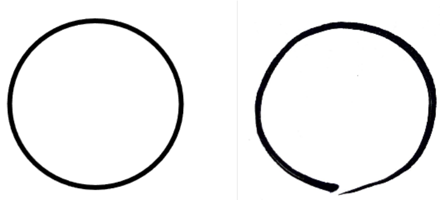


Fig. 1. Example of calligraphy of circles, left) Zen Master Ikyu (Edo period) created a complete circle calligraphy) right) many calligraphy of circles are not complete circle, whichhas a "start point" as breaking point.

1.2. Heart sutra

The Heart Sūtra is a popular sutra in Mahāyāna Buddhism. In Sanskrit, the title Prajñāpāramitāhṛdaya translates as "The Heart of the Perfection of Wisdom" [1]). Below is the first part of the sutra;

[...]
色不異空　空不異色,
色即是空　空即是色,

Corresponding author's E-mail: ysuzuki@nagoya-u.jp.com URL: www.ysuzuki.info

[...]
where 色 means *substances* [2]), 不異, "*not different*" [3]), 空, "*emptiness*" and 即是, "*exactly*" [4]). Hence, an English translation of this part is as follows;

[...]
Substances are not different from Emptiness
Emptiness is not different from Substances
Substances are exactly Emptiness
Emptiness is exactly Substances
[...]

For
Substances are not different from Emptiness and
Emptiness is not different from Substances,

By denoting "*not different from*" as the binary relation R, we obtain binary relations;
[...]
s R e and e R s,
[...]
where s stands for "substances", e, "emptiness"; we call the relation R as H relation, R_H.

Proposition 1 R_H relation is equivalence relation.

The equivalence relation is mathematically satisfied by;
i) xRx (Reflexivity),
ii) xRy and yRx (Symmetry),
iii) xRy and yRz implies xRz (Transitivity).

i) Reflexivity: since substances "are not different from" substances and "emptiness is not different from emptiness," it is obvious (note that R_H stands for the binary relation of "x *not different from* y = x R_H y").

ii) Symmetry: obvious. Because, R_H relation is the statement of the symmetric relation on substances and emptiness.

iii) Transitivity: from i) s R_H s and ii) claims that s R_H e and e and R_H s, if R_H does not satisfy transitivity s R_H e and e R_H s does not fulfill s R_H s, which contradicts to i) and ii). □

Heart sutra is based on H_R and argues that there is no substance in perception with time development. The senses perceive time change; Heart sutra asserts that there is no substance in the perception of such time change. To recognize time change, Heart sutra affirms that time exists. The calligraphy of the circle suggests that symmetry breaking creates time. When time arises, time change is created. We perceive time change; Heart sutra suggests that the perception created by time change has no substance.

2.Method

We consider computations based on the philosophy of Heart sutra; we give basic concepts and notations.

A computing system C is composed of collection of states, S, computing maps, M. We denote the set of states as S and a, b, c, s_i or s_j are element in S; the lowercase of s denotes only one state in S and different suffix denotes different state. M stands for the set of computing maps and for denoting computing maps we will use the lowercase of f, g, h, we call such computing map as "computing map."

A computing map has domain and codomain and transforms from domain to codomain. An arrow denotes a computing map as f: a → b, where a is the domain and b is the codomain; the operation dom(f) gives a, cod(f) gives b. For computing maps f, g, in case cod(f) =dom(g), which is denoted as g ∘ f and called composition f and g; ∘ denotes composition of computing maps. For all computing maps f: a → b, g: b → c, if a computing map Ib gives, Ib ∘ f = f and g ∘ Ib = g, Ib is called identity computing map; we assume that every computing map has identity computing map. Below the word, computing map(s) include(s) identity computing map(s). A computing is a composition of computing maps, where the number of compositions is greater of equal to zero. Things that computing maps can operate are the states of the computing system, while things computing maps cannot operate are not the states of the system.

Definition of Algorithm: A sequence of computing maps from s_i to s_j is called algorithm, where s_i and s_j ; note that an algorithm of f ∘ g and g ∘ f is different, because they are different sequence. We denote sequence of computing maps from s_i to s_j as $s_i \twoheadrightarrow s_j$.
Definition of Programming: Programming is defined as changing the order of the sequences of computing maps.

3.Result

We consider computation without concept of absolute time, quantity and quality that Heart sutra claims.

Let us consider the case where there are more than two computational systems. We assume that the computational systems are autonomous and do not interact with each other, so we will consider a system where more than two computational systems interact with each other.

In the following we will consider the interaction between two computing systems between C_i and C_j. We conjecture those interactions of more than two computing systems requires careful consideration.

Definition of Interaction: Let f_i denote maps of C_i, g_j, of C_j. If the codomain of f_i is contained in the domain of g_j, we define the composition of maps, $g_j \circ f_i$ is the interaction from C_i to C_j. So, $f_i \circ g_j$ are the interaction from C_2 to C_1.

f_i or g_i is a notation for a collection of maps, not only a single map. In $f_i \circ g_k$, f_i and g_k denote collection of maps such that they can compose maps between f_i and g_k. Let C_1 has map m_i: $a \rightarrow b$, m_j: $a \rightarrow c$ and m_e: $j \rightarrow b$; and C_2 has O_i: $b \rightarrow d$, O_k $b \rightarrow h$ and O_p: $a \rightarrow g$. Then f_l is composed of m_i and m_e, because the codomain of m_i and m_e are included by the domain of O_l and O_k. Hence, $f_l = \{m_i, m_e\}$, $g_l = \{O_i, O_k\}$, where {,} denotes a collection of maps.

Definition of Strength of Interaction: We define the magnitude of the overlap between the collection of codomain of f_l and the domain of g_l as the "strength of the interaction". If the overlap is the empty, we define the interaction as 0, or none.

Definition of Inter-Induce Computation, IIC: The composite sequence of maps of C_i and C_j is defined as Inter Induce Computing, IIC or IIC<C_i, C_j>; the case where the composite sequence consists of only the interaction of C_i and C_j is called pure IIC. In this paper, we consider only pure IIC, and for simplicity, we refer to regular IIC as IIC in the following.

We show examples of pure IIC and irregular IIC, below we will omit $\circ$ to denote composition of maps; so $f_i \circ g_j \circ f_m \circ g_n \circ f_k \circ g_k \equiv f_i\ g_j\ f_m\ g_n\ f_k\ g_k$;

regular IIC: $f_i\ g_j\ f_m\ g_n\ f_k\ g_k\ f_m$... ,

irregular IIC: $\underline{f_i\ f_j\ f_m}\ g_n\ f_k\ \underline{g_k\ g_n}$..., underlined compositions illustrate irregular IIC.

Corollary 1 An irregular IIC can be transformed to a regular IIC.

For the irregular sequence $f_i\ f_j\ f_m\ g_n\ f_k\ g_k\ g_n$, by setting F_i $f_i\ f_j\ f_m$ and $G_k = g_k\ k_n$, the sequence is transformed to regular IIC, $F_i\ g_m\ f_m\ G_k$.

Definition of Halt of IIC: When a IIC has zero interaction, the IIC halts. We name the number of compositions of a IIC as length of IIC or composition sequences.

For a collection of maps M_k, we denote the domain and codomain of M_K are respectively, dom(M_k) and codom(M_k).

Corollary 2 In regular IIC of C_i and C_j, if codom(M_i) = dom(M_j) and codom(M_j)=dom(M_i) then IIC does not halt, where M_i and M_j are respectively collection of maps of C_i and C_j.

If the IIC halts, there exist a map h in C_i / C_j such that codom(h) does not in the dom(J) in C_j / Cj. This contradicts the condition of corollary 2.

Proposition 4 In any IIC of C_i and C_j, if codom(M_i) = dom(M_j) and codom(M_j)=dom(M_i) then IIC does not halt, where M_i and M_j are respectively collection of maps of C_i and C_j.

By corollary 1, any irregular IIC can be transformed to regular IIC. And by corollary 2, if the condition of proposition 4 is satisfied, IIC does not halt.

4. Conclusion

In this computational framework, each agent cannot know internal state of others. But, can "implement" algorithm in the interactions. Each agent induces action of others.

We obtained a condition, such interactive computations does not halt, which means, eternal.

Acknowledgement

This research has supported by Grant-in-Aid for Scientific Research No. 21K12108

References

1. https://en.wikipedia.org/wiki/Heart_Sutra (Wikipedia)
2. Bukkyo Jiten (dictionary of Buddhism, in Japanese), (2002), Iwanami Syoten
3. Yukio Sakamoto, (1956), *Soku no igi oyobi kouzou ni tsuite* (*on the meaning and structure of 即 in the Heart Sutra*, in Japanese), Journal of the Japanese Association of Indian and Buddhist Studies, vol.4, (2), 341-350.
4. Jakob Johann Baron von Uexküll, Streifzüge durch die Umwelten von Tieren und Menschen: Ein Bilderbuch unsichtbarer Welten, 1934

Authors Introduction

Dr. Yasuhiro Suzuki

He is an Associate Professor of Graduate School of Informatics, Nagoya University Japan. graduated from Japan Advanced Institute of Science and Technology in 1995, received D. Informatics from Kyoto University in 2001.