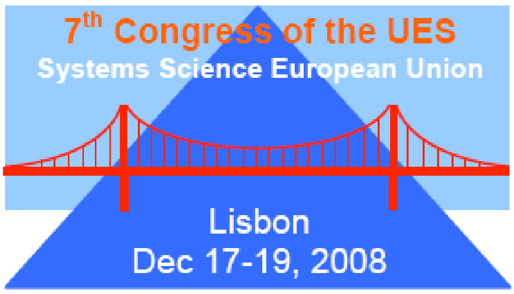
ISBN: 978-972-9059-05-6

the origin/la source : **http://www.afscet.asso.fr/resSystemica/Lisboa08/rato1.pdf**
the title/le titre : **Complexity and emptiness.**
the author/l'auteur : **RATO Raul E. C. T.**
the pages/la pagination : **5 p.**
the year/l'année : **2008**
& the book/la publication: **7th Systems Science European Union Congress Proceedings, Lisboa, Portugal.**

Systems Science European Union
Union Européenne de Systémique
UES

APOCOSIS



Complexity and emptiness

Raul E. C. T. Rato
Uninova
rtr@uninova.pt

Abstract

Along the millennia, complexity has proved to be an elusive concept. Different researchers in diverse fields have worked diligently to produce a wealth of philosophical and theoretical tools to deal with complex phenomena in complex systems. It is known that complexity depends on the observer. Often, there are recognized ”emergent" levels of complexity. The interactions at a lower level of organization result in higher levels with aggregate rules of their own. A defining characteristic of complexity is a hierarchy of description levels, where the characteristics of a superior level emerge from those below it. The condition of emergence is relative to the observer; emergent properties are those that come from unexpected, aggregate interactions between components of the system.
In this paper we take an alternative view. Complexity is found hidden in the simplest concepts and questions like: What’s inside an empty set? This simple question has no simple answer, as the emptiness concept is not void of delicate details. A Hilbert’s selector based formalism is presented as an effort to handle the issue. The classic Russel paradox is analyzed.

Keywords: complexity, empty set, nothing, tolerant, operation, to stuff, to extrude.

Complexity

Along the millennia, complexity has proved to be an elusive concept. Different researchers in diverse fields have worked diligently to produce a wealth of philosophical and theoretical tools to deal with complex phenomena in complex systems. It is known that complexity depends on the observer [1]. Often, there are recognized ”emergent" levels of complexity. The interactions at a lower level of organization result in higher levels with aggregate rules of their own. A defining characteristic of complexity is a hierarchy of description levels, where the characteristics of a superior level emerge from those below it. The condition of emergence is relative to the observer; emergent properties are those that come from unexpected, aggregate interactions between components of the system.

Sets, operations and subtle complexities

Complexity is found hidden in the simplest concepts and questions like: What’s inside an empty set? This simple question has no simple answer, as the emptiness concept is not void of delicate details. In classic set theory a primitive property, primitive in the sense that it is not definable and is assumed to be properly understood by all readers, is the membership property, with $\in$ as symbol. In this theory realm, given any two objects, A and B, either A is a member of B, or it is not a member of B. If this property holds true, then B is a set, and it is said that A is a member of B, or that A belongs to B. This is the fact represented by the expression $A \in B$. If the contrary holds true, then $A \notin B$ is the correct expression, meaning that A is not a member of B, or that A does not belongs to B. A set with a finite number of members is called a finite set, and can be represented listing between braces all its elements. In this theory any object with members is a set. There is a set with no members, called the empty set, the neutral element for the union operation, and it is represented by empty braces $\{\}$, or equivalently by $\varnothing$. No object can belong to the empty set, $\forall x: \; x \notin \{\}$. Any object with no members, but distinct from the empty set, is called an urlement [2]. An urlement is not a set, but can be a member of a set.

In a set, all members must be distinct, and their order of representation is irrelevant. Two sets are deemed equal if they contain exactly the same elements. Accordingly, all the following expressions stand for the same set: $\{a,b,c,a\}$, $\{a,c,b\}$ and $\{a,b,c\}$. The membership property is not transitive, in the sense that given the facts $A \in B$ and $B \in C$, it not permissible to infer $A \in C$. There is a rich and mature theory concerning set manipulation, namely a set algebra with a large plethora of operations like set intersection, set union, set difference, to name a few, whose arguments and results must be sets.

As just seen, set membership is a key property. It is possible to consider the use of membership based operators to manipulate sets. Like the union of sets A and B produces a new set C, $C = A \cup B$, whose elements are all the elements of either A or B, it is possible to conceptualize $C = A \in^{++} B$. In this later expression it is to be understood that the elements of C are all the elements of B, and A itself. Using this "stuffing" $\in^{++}$ operation, A is not considered as a set, but as an element of the resulting set. This is distinct from the union, where the elements of C will be all the elements of B and all of A. As an illustrating example put $A = \{a\,b\}$ and $B = \{b\,c\}$. Then $A \cup B = \{a\,b\,c\}$, but $A \in^{++} B = \{\{a\,b\}b\,c\}$, an outcome well distinct from the former one. And $\in^{++}$ is valid when its first argument any object of the theory, a set or an urelement. The union operation, $\cup$, requires both arguments to be sets. To transform an urlement into a set, we need to use $\in^{++}$. There is no operation in classic set algebra to transmute a single urlement, like a, to the matching singleton set, $\{a\}$. Using this stuffing operation, it is possible to stuff any urelement into the empty set, as in $\{a\} = a \in^{++} \{\}$. Its is also possible to stuff sets into sets, as in $\{\} \in^{++} \{\} = \{\{\}\}$. The $\in^{++}$ can only stuff sets. Stuffing something into an object that is not a set, will output the unchanged object. Stuffing a set into itself will produce a new set. This is coherent with the foundation axiom, which explicitly forbids a set to be member of itself. This can be expressed as: $A \in^{++} A = A \quad iif \quad A \; is \; not \; a \; set$

The quest for the neutral nothing

It is well-known that the empty set is the union operation neutral element. It is possible to ask for the neutral element of this $\in^{++}$ stuffing operation. This neutral stuffing element is not the empty set. As seen, if a set is stuffed with the empty set, the result is yet another set. What kind of "thing" can be stuffed into an empty set, and yield the empty set itself as result? Inside an empty set there is nothing. To stuff nothing into an empty set should do the trick. To ease the manipulation details, let us define $\varnothing$ as the symbol for nothing. Clearly, to establish a conventional sign to nothing doesn't turn it into something, specifically into a set theory object. This symbol looks like zero, but it's not zero. Looks like the empty set, but it's not the empty set. And we have $\varnothing \in^{++} A = A$, for any A. And an important property emerges: no set has $\varnothing$ as member, $\varnothing \notin A, \forall A$.

What is $\varnothing$? It is just a practical way to represent the void, the nothing, the emptiness. It is not a set theory object. As a matter of fact, it stands for the non-object. As seen, lets call it the $Nothing$.

Inside any set, there can be plenty of $Nothing$, as in $\{a \varnothing b \varnothing\} = \{a\,b\}$.

The number of elements of a set with only $Nothing$ inside is zero. Although somehow related, zero and $Nothing$ are not the same. Zero is a number. $Nothing$ is nothing. $Nothing$ is the $\in^{++}$ neutral element. Zero is the sum neutral element. When numbers were represented by tally notches, zero was the no notches tally. A tally with zero notches is something. Similarly, the ancient Babylons used no special symbol for zero, just an empty space, a place holder. Brahmaguptha went further, giving it a symbol in order to deal with its operating capabilities. That was necessary because to make operations possible, symbols are needed [3].

The quest for the $\varnothing$ operating capabilities can be based on the properties of a set dyadic's operations. On a set, a finite dyadic operation can de defined by a 2D table, built solely with the set elements. If some of those elements don't show up as table lines or columns titles, or if some table entries are left undefined, the operation is not completely defined. If all titles are set, and for each table entry pair (first the line, then the column) the outcome is stipulated, the operation is completely defined, and it is also known as a binary operation. The general binary operation symbol will be f_n.

Lets us consider now how to operate with $\varnothing$. From the "operations as tables" perspective this symbol is just another possible table entry/outcome. A tolerant operation is a $\varnothing$ enabled dyadic operation. Put $A = \{a\}$. On this set, it is possible to define only one binary operation, as depicted in this table:

f_0	a
a	a

To turn this operation into a tolerant one, it is necessary to deal with the $\varnothing$ symbol. Without changing the input description, another operation can be defined, as depicted in this table

f_1	a
a	$\varnothing$

But the $\varnothing$ symbol can show up as a line title, a column title, or both, as in

	a
$\varnothing$	
a	

	$\varnothing$	a
a		

	$\varnothing$	a
$\varnothing$		
a		

The last case, were $\varnothing$ can show up anywhere in the table is the most general one. Such a binary operation is called a tolerant binary operation, or simply a tolerant operation, and its general symbol will be $\widehat{f}_n$. In this singleton case it is possible do define sixteen distinct tolerant operations. The following table depicts $\widehat{f}_7$, assuming the standard natural binary numbering convention under the bijective recoding $0 \leftrightarrow \varnothing, 1 \leftrightarrow a$:

$\widehat{f}_7$	$\varnothing$	a
$\varnothing$	a	a
a	a	$\varnothing$

As shown, tolerant operations of the singleton set case can be put in a one-to-one correspondence with the standard sixteen bi valued Boolean two inputs/one output functions.

Classical logic deals with the concept of trueness, recognizing only two possibilities. Bi valued Boolean algebra is its symbolic counterpart. Many-valued logics are non-classical logics. One example of this is tri valued logic. It is not possible to develop a Boolean algebra counterpart for tri valued logic [4]. This logic as seen some practical applications, namely in the popular SQL database language, where the truth values can be $True$, $False$ and $Null$. Equating $Null$ with $\varnothing$, it is possible to show that tri valued logic is nothing else that classic logic with tolerant operations enabled.

Inverting $\in^{++}$

Using the Hilbert selector, ι, it is possible to dig up one of the elements of a set [4, Santos Guerreiro]. If A is a non empty set, then ιA is one of its elements. This ι always selects the same element, so ιA stands always for the same element, as long as the set A is still the same. A similar, but $\varnothing$ enabled device is $\in^{--}$. Applied to a non empty set, it extrudes of one of its elements, and the residual set cardinality will drop by one, a clear distinction from the ι. Also in contrast with ι is the fact that it is "memoryless", in the sense that if applied to several copies of the same set, each time the extruded element can be a diverse one. By definition, it is not possible to know in advance anything that hints what the extruded element will be. If applied to an empty set its output will be $\varnothing$, as expressed in $\in^{--}\{\}=\varnothing$. As an illustrating case, $\in^{--}\{a,b,\alpha\}=b$, and the residual set for this case will be $\rho(\in^{--}\{a,b,\alpha\}=b)=\{a,\alpha\}$. This extruder works only with sets. When applied to a non set object, its output will be the unchanged object. This assures the following inverting property:

$$(\in^{--}A)\in^{++}A=A,\forall A$$

Final considerations

Let us consider again the expression $A\in^{++}A$. As seen, this stuffs the set A into itself, giving rise to a new set. Lets stuff this new set with itself: $(A\in^{++}A)\in^{++}(A\in^{++}A)$, simplified to $(A\in^{++}A)^2$. Let us stuff it to itself once more. We get

$((A\in^{++}A)\in^{++}(A\in^{++}A))\in^{++}((A\in^{++}A)\in^{++}(A\in^{++}A))$, abbreviated to $(A\in^{++}A)^3$. In this way it is possible to define $(A\in^{++}A)^n$, for any positive integer n. Lets deal now with Ω, the so called set of all sets. Clearly we can put $(\Omega\in^{++}\Omega)^n$, and obtain new sets, which are not members of Ω. This shows that to speak about the set of all sets is as senseless as to speak of the natural number greater that all the others. In this way the Russel paradox never shows off.

Acknowledgements

The author wishes to express his thanks to the many colleagues and to doctor Duarte-Ramos both for the stimulus to write this paper and for many valuable suggestions.

References

[1] Bruce Edmonds, « Syntactic Measures of Complexity», *Doctoral Thesis*, University of Manchester, Manchester, 1999.

[2] Patrick Suppes, *Axiomatic Set Theory*, Dover Publications, New York, 1972.

[3] Karl Menninger, *Number words and number symbols*, Dover Publications, New York, 1992.

[4] Carlos Sêrro, *Sistemas digitais: Fundamentos algébricos*, IST Press, Lisboa, 2003.

[4] J. Santos Guerreiro, *Curso de Análise Matemática*, Escolar Editora, Lisboa, 2007.