International Journal of Computational Intelligence Systems, Vol.3, No. 2 (June, 2010), 148-159

# What is Soft Computing? Revisiting Possible Answers

**L. Magdalena**

*European Centre for Soft Computing*
*C/ Gonzalo Gutiérrez Quirós*
*Mieres 33600, Asturias (Spain)*
*E-mail: luis.magdalena@softcomputing.es*

Received: 30-12-2009
Accepted: 28-05-2010

**Abstract**

The term Soft Computing was coined by L.A. Zadeh in the early 90's. Since that time many researchers have tried to define it considering different approaches: main constituents, properties, abilities, etc. In addition, the term Computational Intelligence has also gained popularity having a somehow quite close meaning to that of Soft Computing. The central idea of this paper is to present, analyze, compare and discuss a few of the definitions that can be found on literature; not trying to find the best but to offer the reader arguments to make his/her own decision.

*Keywords:* Soft Computing, Computational Intelligence, Fuzzy, Neural, Evolutionary, Probabilistic, Hybridization

## 1. Introduction

"What is Soft Computing?" is likely to be the first question you will be asked by an outsider once you reveal that you work in Soft Computing (SC) or that you apply SC techniques. Nevertheless, this can be a startling question, and a difficult one to answer, for a researcher or a practitioner in Soft Computing: if you are familiar with the field, you do not ask such a question yourself and thus you may not have a clear cut definition available. Of course, there is also the additional problem that, as a term of natural language, *Soft Computing* lacks a precise and generally accepted definition. Furthermore, the field that has been labeled *Soft Computing* is fairly new and thus still evolving rapidly, making it difficult, if not impossible, to provide a final answer.

In addition, the term *Computational Intelligence* has also gained popularity in recent years. Having according to some researchers a quite close meaning to that of SC, others consider there are clear differences between them two. So, the coincidences and discrepancies in between SC and Computational Intelligence (CI) should also be considered when trying to define Soft Computing.

Furthermore, being SC heavily involved in commercial and industrial applications produces a continuous exposure and interaction of people working on SC with people not familiar to the topic. An interaction where the question "What is Soft Computing?" is always present. For sure this question has neither a single nor an easy answer. Nevertheless, the simple fact that the question is commonly asked, forces us to spend some thought on apt and comprehensible answers.

Researchers have defined Soft Computing in terms of properties of its constituent technologies, by enumerating the technologies, by comparison with other concepts, or considering its use. That means several definitions from different points of view (many others could be added), however, none of them can be called the "correct" or "best" answer: depending on the context, the background of the person asking the question and other factors, each of them may be the most appropriate to give. The central idea of this paper is to present, analyze, compare and discuss a few of the definitions that can be found in the literature; not trying to find the best but to offer the reader arguments to make his/her own decision.

From this point, the paper explores several answers that have been suggested in the past, each of which uses a different approach and emphasizes different aspects. These include definitions and characterizations of SC

- as a family of techniques,
- by essential properties,
- as a complement of hard computing, or
- as a tool for coping with imprecision and uncertainty.

As previously said, it is also interesting to consider other terms closely related to SC as is the case of *Computational Intelligence*, and analyze the level of intersection or even the coincidence between these two concepts.

Before starting with any definition it could be quite useful to put into context the genesis of the idea according to the view of its father, Lotfi Zadeh[1]:

> The concept of soft computing crystallized in my mind during the waning months of 1990. Its genesis reflected the fact that in science, as in other realms of human activity, there is a tendency to be nationalistic – to make an exclusive commitment to a particular methodology and proclaim that it is superior to all others. It is this mentality that underlies the well-known hammer principle: when the only tool you have is a hammer, everything looks like a nail.

Being the result of the consideration on the difficulties generated when following the "hammer principle", SC represents a rejection of this mentality. This is a key aspect to be considered when describing it, as will be shown in next section.

## 2. Soft Computing as the mixture of several preexisting techniques

The term Soft Computing was coined to refer to a family of computing techniques particularly adapted to cope with a class of problems for which other techniques were not quite well suited. Zadeh[2] established that:

> Basically, soft computing is not a homogeneous body of concepts and techniques. Rather, it is a partnership of distinct methods that in one way or another conform to its guiding principle.
>
> ...
>
> The principal constituents of soft computing are fuzzy logic, neurocomputing, and probabilistic reasoning, with the latter subsuming genetic algorithms, belief networks, chaotic systems, and parts of learning theory. In the partnership of fuzzy logic, neurocomputing, and probabilistic reasoning, fuzzy logic is mainly concerned with imprecision and approximate reasoning; neurocomputing with learning and curve-fitting; and probabilistic reasoning with uncertainty and belief propagation.

We can imagine this concept as a first step into solving the restrictions described by the "hammer principle", done by replacing the "hammer" by a complete toolbox, and this first definition includes a sort of enumeration of the available tools in our toolbox.

Another lesson from this enumeration is the fact that SC is an evolving concept. In this early definition, genetic algorithms are included as part of probabilistic reasoning, but, as the presence of genetic algorithms has gained importance in the field, more recent enumerations have included evolutionary computation (EC, a family of search and optimization techniques that among others include genetic algorithms) as an independent constituent of soft computing. In that sense, most enumerations of SC components included Fuzzy, Neural, Probabilistic and Evolutionary Computation as the four basic components.

This evolving character of SC when defined as the mixture of several techniques produces as a consequence a certain absence of clear and fixed borders. This situation generates some criticism and even some controversies when considering if a certain technique is or is not part of SC. On the other hand, the absence of clear borders is surely the best way to overcome the *hammer principle* by incorporating to our toolbox any new tool serving to our needs. In any case, it is clear that simply defining SC as a collection of techniques, when this collection could be continuously changing, is not enough. A first step forward is that of considering SC not only as a collection of several techniques put side by side, but as a family of tightly interacting techniques. In that way appeared the term Hybridization playing a central role in SC.

Although Fuzzy, Neural, Probabilistic and Evolutionary techniques (the traditional constituents of SC) share some common characteristics, they are considered complementary as desirable features lacking in one approach are present in another. Consequently, after a first stage in which they were applied in isolation, the last decade of the past century (that of the appearance of the term SC) witnessed an increasing interest on hybrid systems obtained by symbiotically combining those four components. The term "Soft Computing" and the interest on hybrid system appeared and has grown almost simultaneously. We can even say that the term Soft Computing makes no sense without the concept of Hybrid Systems.

### *2.1. Soft Computing and Hybridization*

As already mentioned, the idea of SC as the mixture of several preexisting techniques goes beyond putting different techniques in a single toolbox. The mixture includes symbiosis in between techniques, being for many researchers one of the most significant features of SC.

In this sense one of the main characteristics of SC is that of hybridization. Hybrid approaches could be considered as one of the main contributions of SC, with neuro-fuzzy systems being the first and probably the most successful hybrid approach till now.

Neuro-fuzzy systems incorporate elements from Fuzzy Logic (FL) and Neural Networks (NN). This idea of hybridization originates from two observations:

1. Fuzzy Systems are neither capable of learning, adaptation or parallel computation, whereas these characteristics are clearly attributed to NNs.

2. NNs lack flexibility, human interaction, interpretability or knowledge representation, which lies at the core of FL.

Similar arguments were used latter to generate other hybrid approaches like Genetic Fuzzy Systems, where the main aim was to join the robust search capabilities of EC, and the already mentioned properties of FL.

From this point of view, SC, that started as the *partnership of fuzzy logic, neurocomputing, and probabilistic reasoning*, has evolved to integrate many symbiotic approaches among those techniques. This idea of hybridization, being central in SC, is reflected by different authors.

As an example, the following quotation[3] gives a clear view of the role of hybridization

> SC's main characteristic is its intrinsic capability to create hybrid systems that are based on a (loose or tight) integration of constituent technologies. This integration provides complementary reasoning and searching methods that allow us to combine domain knowledge and empirical data to develop flexible computing tools and solve complex problems.

Figure 1, recreated from the cited paper, gives a graphical view of this concept. Many hybrid approaches appear at bottom of the four boxes representing the main constituents of SC (the arrows show the origin of the hybridized components).

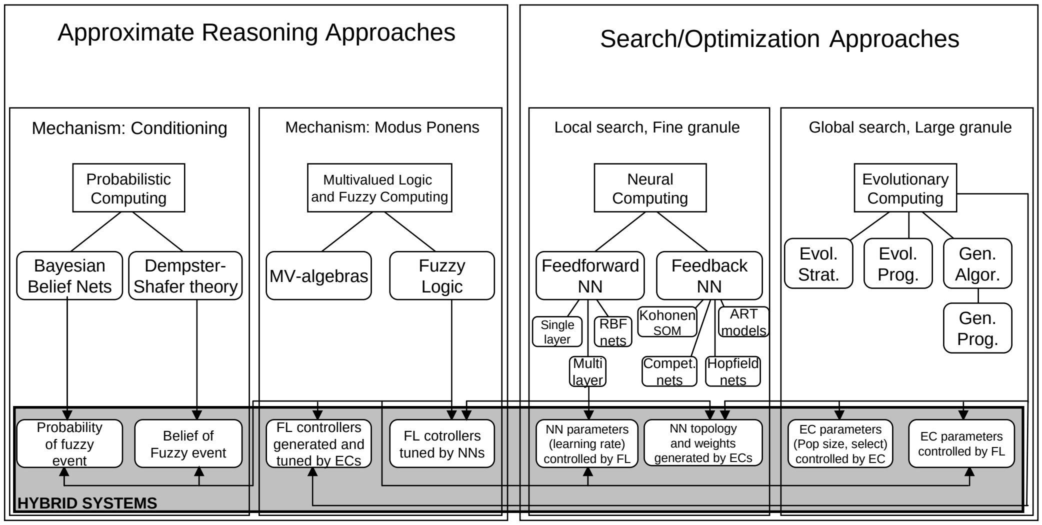


Fig. 1. Hybridization in Soft Computing according to Bonissone et al.[3].

Some examples are the “FL controllers tuned by NNs”, where the originating components are Fuzzy Logic and Multilayer Perceptron, or the “NN topology and weights generated by ECs” obtained as a result of merging Multilayer Perceptrons and Evolutionary Computing.

This central role of hybridization is also represented by Figure 2, from Cordón et al.[4], through the idea of different *hybrid approaches* appearing as the intersecting areas of the *main components* of SC.

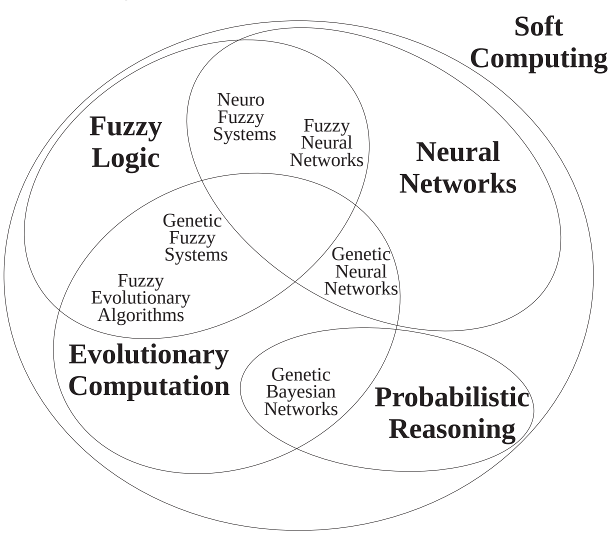


Fig. 2. A graphical view of Hybridization according to Cordón et al.[4].

In summary, SC has been defined as the mixture of several preexisting techniques, but pointing out the importance of the interaction in between those techniques, formalized in terms of different hybrid approaches.

## 3. Soft Computing defined by essential properties

As previously stated, the definition merely based on the enumeration of constituent seems not being enough. So, it makes sense to consider the properties or characteristics of those constituents in order to determine what, if any, are the common (essential) properties of SC.

A previous discussion on “What is SC?” took place as part of the FLINS’96 conference[5]. From that article has been taken the following definition of SC:

> Every computing process that purposely includes imprecision into the calculation on one or more levels and allows this imprecision either to change (decrease) the granularity of the problem, or to “soften” the goal of optimization at some stage, is defined as to belonging to the field of soft computing.

It is clear that this is a quite general definition including all those techniques and approaches that have been previously described as the components of soft computing. Consequently both definitions seem to be in accordance.

The key idea in this point of view is that SC *purposely includes imprecision into the calculation* changing (relaxing) the level of description of the problem or the level of achievement of the goal. But what should be clearly stated is that the imprecision is not a target, but a mean or even a need to achieve a higher order objective that could be summarized as "solvability" of the problem, and will be considered again in other definitions of SC. If we go further in this approach, it is possible to say that, in summary, soft computing encompasses two main conceptual components, namely approximate reasoning and function approximation and optimization. This view was already present in Figure 1, where the corresponding techniques are located in two main boxes referring to those two conceptual components.

Approximate reasoning is considered when relaxing the level of description of the problem (*decrease the granularity*) while function approximation and optimization is considered by relaxing the level of achievement of the goal (*soften the goal of optimization*).

In summary, we can say that SC techniques are those reasoning and optimization techniques that assume and integrate imprecision when solving a problem either as a consequence of the presence of that imprecision in the available information or as a mean to overcome complexity and achieve feasibility/solvability. From that point, and considering again that soft computing is an evolving concept, its evolution can be easily interpreted in terms of the integration of new topics/techniques that are in accordance with these properties and the corresponding conceptual components.

In that sense we can mention *granular computing*[6] as one new and important technique to be integrated in SC. Granular computing clearly matches with the idea of relaxing the level of description of the problem or the model in order to achieve suitable solutions to complex problems. As a matter of fact, granular computing encompasses concepts from (among others) rough sets and fuzzy sets theories, both of them considered as constituents of SC.

## 4. Soft Computing as opposite to Hard Computing

The term *soft computing* distinguishes those previously enumerated techniques from *hard computing* (conventional approaches) considered as less flexible and more computationally demanding. The key aspect for moving from hard to soft computing is the observation that the computational effort required by conventional approaches which makes in many cases the problem almost infeasible, is a cost paid to gain a precision that in many applications is not really needed or, at least, can be relaxed without a significant effect on the solution. In other words, we can assume that this imprecision is a small price to be paid for obtaining more economical, less complex and more feasible solutions.

To clarify that, it would be easier to talk in terms of optimization, and the situation is that, for many different applications a sub-optimal solution is enough, and having that in mind when designing the optimization process will provide the difference between obtaining a solution that satisfies our needs or getting lost when searching for the optimal solution. It is again the idea of *softening the goal of optimization* taken from Li et al.[5] and presented in previous section.

A different but quite related point of view is that stated by Kercel[7]:

> Actually, the distinguishing feature of soft computing is straightforward. Hard computing uses an explicit model of the process under consideration while Soft computing does not do this. Instead, as an indispensable preliminary step, it infers an implicit model from the problem specification and the available data.

Working out the analogy between both previous paragraphs we can see that building the explicit model is a first step in the process of finding the optimal solution, while, in absence of such an explicit model, the use of an implicit model usually drives us to a (sub-optimal) solution satisfying our needs. Considering again the view of SC as blending approximate reasoning and function approximation/optimization, this idea concentrates on the second part, the one usually assigned to neural networks and evolutionary computation, that could obviously integrate other optimization techniques as new components of soft computing.

In the same way that evolutionary computation gained some presence in the field, meriting for a soloist role in the ensemble (and not a secondary role as occurred in some initial definitions), other new bio-inspired search and optimization techniques have appeared on stage more recently to play similar roles as EC. And even heuristic techniques that emerged inspired by the principle that satisfaction is better than optimization could be considered as part of soft computing[8].

In any case, some limits are needed since function approximation and optimization is a large area not completely contained in SC, e.g., numerical optimization techniques are clearly out of the scope of SC if we consider the topics of interest of any conference or publication in the field.

A different approach to the same idea of Soft Computing as opposite to Hard Computing was stated by Bonissone[9] in terms of Approximate vs. Precise models, instead of Implicit vs. Explicit models:

> In ideal problem formulations, the systems to be modeled or controlled are described by complete and precise information.
>
> ...
>
> As we attempt to solve real-world problems, however, we realize that they are typically ill–defined systems, difficult to model and with large–scale solution spaces. In these cases, precise models are impractical, too expensive, or non–existent. The relevant available information is usually in the form of empirical prior knowledge and input–output data representing instances of the systems behavior.

Figure 3 has been adapted from this reference by also including the idea of Explicit and Implicit models (from Kercel[7]) to integrate both conceptions. The figure clearly states the role of Soft Computing as the machinery to cope with problems that are out of the scope of the more traditional Hard Computing techniques. In fact the figure describes SC as the way of approaching problems when a precise model is either unavailable or unaffordable due to its complexity.

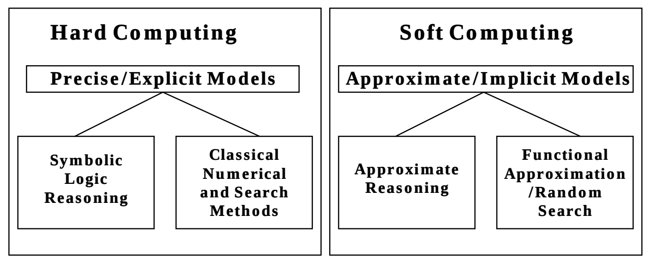


Fig. 3. Hard and Soft Computing

## 5. Soft Computing as a tool to cope with uncertainty and imprecision

SC techniques are meant to operate in an environment that is subject to uncertainty and imprecision. According to Zadeh[2], the guiding principle of soft computing is:

> exploit the tolerance for imprecision, uncertainty, partial truth, and approximation to achieve tractability, robustness, low solution cost and better rapport with reality.

Imprecision results from our limited capability to resolve detail and encompasses the notions of partial, vague, noisy and incomplete information about the real world. In summary, SC is designed to cope with all those "uncomfortable characteristics" that real world uses to add to the problems under consideration. In other words, it becomes not only difficult or even impossible, but also inappropriate to apply hard computing techniques when dealing with situations in which the required information is not available, the behavior of the considered system is not completely known or the measures of the underlying variables are noisy.

The point of view described in previous paragraphs is the most common approach when talking about SC as a tool to cope with uncertainty and imprecision, but we can go one step further and analyze those situations where *imprecision* is not a drawback of the information we are managing but an intrinsic characteristic of that information. In that sense it is quite important to distinguish between measurements and perceptions, and to discriminate those situations where we work on the basis of imprecise measurements from those others where we *compute with perceptions*. Computing with perceptions is clearly an approach to approximate reasoning (again one of the constituents of soft computing).

In that sense, soft computing as a tool to cope with uncertainty and imprecision should clearly include computing with words and perceptions as one of its basic components. In fact, quite similar sentences to that previously quoted are used by Zadeh[10] to describe the potential use of *computing with words* as well as its *computational theory of perceptions*. Computing with words is inspired by the remarkable human capability to perform a wide variety of physical and mental tasks without any measurements and any computations. As a methodology, computing with words provides a foundation for a computational theory of perceptions.

A basic difference between perceptions and measurements is that, in general, measurements are crisp whereas perceptions are fuzzy. One of the fundamental aims of science has been and continues to be that of progressing from perceptions to measurements. One of the potential achievements of soft computing will be to return to perceptions in order to take profit of its qualities as a carrier for intrinsically imprecise information.

## 6. Soft Computing according to the use

Finally, to somehow determine what is the common use of "Soft Computing" we will consider Scopus bibliographical data base and will perform a search using different terms. The idea is to obtain a first view of the importance of Fuzzy, Neural, Evolutionary and Probabilistic concepts on the overall "toolbox".

The first analysis will consider the appearance of the term "Soft Computing" in the *Scopus* data base. We will count only those articles in which this term appears as part of the title, abstract or keywords. Then, we will search (again in title, abstracts or keywords) for terms related to the main techniques in the field. Each of the four concepts will be represented by one or several terms. In the case of *Fuzzy Logic* we will consider only the term "fuzzy", for *Neural Networks* we will use the terms "neural", "neuro", and "neuronal"; *Evolutionary Computation* will be represented by "evolutionary", "genetic", "evolution", and "evolving"; and finally, the search for *Probabilistic Reasoning* (PR) will consider the terms "probabilistic", "probability", "bayes", and "bayesian". The selection of terms is the result of some refinements over previous searches but could surely be improved. In any case the idea is to have a rough picture of the field and not to make a perfect portrait. All searches were performed on Dec. 19th, 2008 to get a fixed picture.

Table 1. Articles in Scopus containing the term "Soft Computing".

| | SC | SC & FL | SC & NN | SC & EC | SC & PR | CI & none in abs. | in art. |
|---|---|---|---|---|---|---|---|
| 2008 | 379 | 160 | 151 | 100 | 33 | 102 | 35 |
| 2007 | 335 | 146 | 128 | 96 | 25 | 93 | 37 |
| 2006 | 288 | 137 | 122 | 72 | 24 | 70 | 37 |
| 2005 | 247 | 123 | 101 | 69 | 17 | 61 | 28 |
| 2004 | 232 | 145 | 98 | 73 | 27 | 43 | 22 |
| 2003 | 158 | 105 | 81 | 66 | 13 | 22 | 8 |
| 2002 | 121 | 75 | 66 | 46 | 11 | 23 | 9 |
| 2001 | 149 | 108 | 103 | 71 | 12 | 17 | 6 |
| 2000 | 104 | 61 | 63 | 44 | 6 | 20 | 15 |
| 1999 | 75 | 55 | 43 | 35 | 7 | 7 | 2 |
| 1998 | 56 | 43 | 27 | 26 | 3 | 8 | 4 |
| 1997 | 36 | 24 | 16 | 11 | 3 | 10 | 5 |
| 1996 | 38 | 30 | 23 | 17 | 3 | 6 | 2 |
| 1995 | 28 | 22 | 16 | 10 | 4 | 4 | 4 |
| 1994 | 9 | 7 | 5 | 1 | 2 | 2 | 1 |
| 1993 | 3 | 3 | 3 | 1 | 1 | 0 | 0 |
| Total | 2258 | 1244 | 1046 | 738 | 191 | 488 | 215 |

Table 1 compiles the information structured by years to give us also an idea of evolution in time. First column refers to the number of papers containing the term "Soft Computing". Then, four columns analyze the simultaneous presence of that term with those other terms chosen to characterize the presence of each of the four main techniques of SC. In all the cases we consider only the title, abstract and keywords. The two last columns compile those articles that, including the term *soft computing*, do not contain any of the other terms related to fuzzy, neural, evolutinoary or probabilistic approaches, previously cited: the first column considering only title, abstract and keywords; while the second one after searching all through the article.

Considering only the title, abstract and keywords in a first stage tries to get a clearer view of the main topic of the article. Adding later the body of the article to get the number of papers not mentioning the main techniques of SC is basically related with the fact that many reviews, states of the art and similar papers do not mention fuzzy, neural, evolutionary or probabilistic in the abstract, but clearly consider them in the body of the article.

A first analysis of the table says that almost 80% of the articles refer to one or more of the main techniques in title, abstract of keywords; and more than 90% refer to it along the whole paper.

A second interesting question is the evolution in time. As an example, FL which started with figures over 70% in the early years, moves in the range of 40–50% in the last four years. NN has moved from figures around 60% to values close to 40%. EC has slightly reduced is percentage while PR maintains quite similar values all through the period. In what concerns papers not mentioning the techniques, its ratio has slightly grown in recent years (around 25%) if only considering title, abstract and keywords; but if we consider the whole paper, the ratio is almost constant around 10%.

It could be also interesting to consider what is the distribution of different intersections among techniques, or at least, what techniques are mentioned together in the articles. To do so, Figure 2 previously used to depict a graphical view of hybridization placing hybrid approaches at the intersection of the corresponding techniques, has been recreated in Figure 4 by replacing the qualitative intersections by quantitative intersections. In that sense, the qualitative intersection in between fuzzy logic and neural networks was represented in the former figure by neuro fuzzy systems and fuzzy neural networks. Now, the quantitative intersection will be described by the number of papers simultaneously containing the terms related to fuzzy and neural approaches, i.e., the intersection between second and third columns in table 1.

Figure 4 defines the map of all possible intersections considering the four techniques. It is not easy to show all possible intersections among four sets in a two-dimensional picture. The natural way was a 3-D view using a sort of tetrahedron and putting one set on each of the four vertex, but the number of different intersections makes it difficult to transfer that figure to a two-dimensional view. To solve the situation we have started with only three sets (we can assume that are the sets located at the three vertex of the triangle being the base of the tetrahedron) being fuzzy, neural and evolutionary techniques (including all potential intersections out of probabilistic). Now, it is impossible to add all items (papers) containing probabilistic techniques as a single set intersecting all three others. So, this set has been added in two parts: papers containing probabilistic and evolutionary terms (small circle labeled PR & EC, inside EC), and those containing PR but not EC terms (the circle labeled PR). In that way we get in a single picture all possible combinations among the four sets (NN, EC, FL, and PR), including papers considering a single technique, as well as those referring to two, three or the four techniques.

From this picture we can determine the exact distribution of the 1770 papers containing (in title, abstract or keywords) the term "Soft Computing" and, at least, one term related to FL, NN, EC or PR (1770 is obtained from the overall number of papers, 2258, minus 488 papers which did not mention any technique in title, abstract or keywords). As an example we can see that 36 of those papers mention the four techniques; 295 mention FL, NN and EC (but not PR); 346 mention FL and NN (not EC or PR); and finally, 366 mention uniquely FL.

If we order the techniques (and combinations) according to the number of papers referring to it, the top five are: FL, FL&NN, FL&NN&EC, NN, and EC. Another interesting point is that approx. 60% of the papers mention more than one technique demonstrating the large presence of hybridization in the field. In addition, any of the four techniques has a larger presence in combination than alone.

So, according to the use, we can say that the core techniques of SC are fuzzy, neural and evolutionary computation; with a quite important presence of hybrid systems.

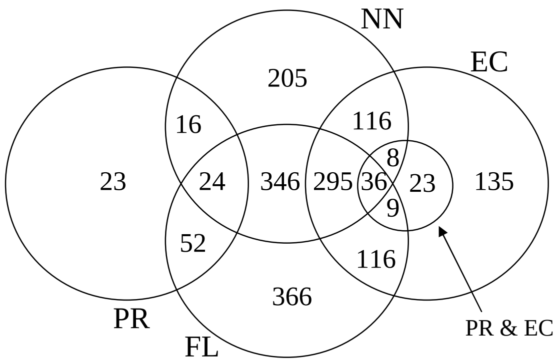


Fig. 4. Number of articles according to mentioned techniques.

## 7. Soft Computing and Computational Intelligence

Once considered many different ways to define SC, another interesting question is its relation with other terms as "Computational Intelligence". Is there any difference between Soft Computing and Computational Intelligence? Are simply different terms for the same topic? To have a better view of that question let us start by considering also some definitions of Computational Intelligence (CI).

We can follow a similar sequence that the one considered for SC, i.e., definition by essential properties, as a family of techniques or as complementary to something else. It would be much more difficult to find a parallel structure to that of Soft Computing as a tool to cope with uncertainty and imprecision, maybe that will be a difference between the two terms.

Following the same order previously considered we will start with a definition based on the component technologies and their essential properties, and in that sense we can directly consider the one applied by the Computational Intelligence Society of IEEE, a society using the slogan "Mimicking nature for problem solving". A slogan that is clearly reflected in the following statement (from `http://ieee-cis.org/about_cis/scope/`):

> The Field of Interest of the Society shall be the theory, design, application, and development of biologically and linguistically motivated computational paradigms ...

From this statement we can deduce that CI could be defined as the area of research that studies biologically and linguistically motivated computational paradigms, having apparently nothing to do with the corresponding definition of SC. But only the first part of the statement was included in the previous quotation, let us now analyze the second part:

> ... emphasizing neural networks, connectionist systems, genetic algorithms, evolutionary programming, fuzzy systems, and hybrid intelligent systems in which these paradigms are contained.

And what we find now is quite similar to the definition of SC as a family of techniques we considered previously. And what is even more interesting is the explicit reference to hybridization, another distinctive characteristic of SC. It is also possible to recover other citations using the same approach of definition by components, as in the following one by Pedrycz[11].

> Computational intelligence is a recently emerging area of fundamental and applied research exploiting a number of advanced information processing technologies. The main components of CI encompass neural networks, fuzzy set technology and evolutionary computation.

Consequently, if we approach CI from the point of view of components the result is quite close, maybe we miss in the case of CI the field of Probabilistic Reasoning, a topic that had an important role in SC. This absence is also supported by the first definition, since (at least in a first view) Probabilistic Reasoning is neither biologically nor linguistically motivated.

Continuing the parallel description of SC and CI we find now the idea of "complementary or opposite to something else". In that sense, while SC was defined as based on approximate or implicit models being somehow a complement to Hard Computing, the following quotation taken from Duch[12], locates Computational Intelligence in a quite similar position:

> Computational Intelligence is a branch of science studying problems for which there are no effective computational algorithms.

This statement could be read as expressing the idea of CI being the branch of science considering those problems for which there is not an exact model, plus those cases where the model exist but its consideration is not computationally effective, i.e., *when we need to reduce the granularity or soften the goal*. Both ideas where presented previously as descriptions of SC as the opposite to hard computing or based on its essential properties.

So, apparently there is no significant difference in between Soft Computing and Computational Intelligence. It would be interesting to check for the presence of the term "Computational Intelligence" and the related techniques in the same data base (Scopus) previously considered. Using the same search patterns that for Table 1, but replacing SC with CI, Table 2 has been generated.

Table 2. Articles in Scopus containing the term "Computational Intelligence".

| | CI | CI & FL | CI & NN | CI & EC | CI & PR | CI & none in abs. | in art. |
|---|---|---|---|---|---|---|---|
| 2008 | 261 | 75 | 91 | 26 | 21 | 88 | 24 |
| 2007 | 772 | 117 | 150 | 33 | 50 | 435 | 300 |
| 2006 | 235 | 56 | 76 | 19 | 15 | 89 | 32 |
| 2005 | 166 | 64 | 75 | 32 | 9 | 48 | 18 |
| 2004 | 112 | 38 | 67 | 22 | 8 | 23 | 13 |
| 2003 | 69 | 25 | 40 | 18 | 4 | 18 | 11 |
| 2002 | 61 | 24 | 28 | 13 | 2 | 24 | 7 |
| 2001 | 54 | 29 | 31 | 15 | 1 | 8 | 4 |
| 2000 | 34 | 18 | 17 | 11 | 2 | 8 | 5 |
| 1999 | 38 | 24 | 28 | 19 | 2 | 4 | 6 |
| 1998 | 42 | 15 | 17 | 5 | 1 | 14 | 9 |
| 1997 | 38 | 18 | 26 | 10 | 2 | 7 | 5 |
| 1996 | 17 | 9 | 11 | 2 | 1 | 3 | 2 |
| 1995 | 16 | 9 | 12 | 6 | 1 | 2 | 2 |
| 1994 | 8 | 4 | 6 | 4 | 0 | 2 | 2 |
| Total | 1923 | 525 | 675 | 234 | 119 | 775 | 438 |

But before analyzing the figures in this new table, it is important to consider the case of the year 2007 that is completely out of the series. A simple check shows that from 772 articles in 2007, 449 proceed from a single conference plus one associated workshop (from those papers 325 do not include the terms related to fuzzy, neural, evolutionary or probabilistic techniques in the title, abstract and keywords; and 251 of them do not mention any of those terms all through the paper). To reduce the effect of this unusual year, mostly related to a single conference, 2007 will not be considered when computing the distribution of papers.

By comparing the figures from both tables we can recognize some similarities. In both cases 90% of the papers contain concepts related to fuzzy, neural, evolutionary or probabilistic. The presence of neural (around 45% in both cases) and probabilistic techniques (8 and 6%) are quite similar in papers refering to SC and CI. This point could be interesting since PR was apparently not considered as one of the core technologies of CI while the figures show a similar presence than that in SC. The presence of fuzzy and evolutionary techniques is lower in the case of CI (from 55 to 35% in fuzzy and from 33 to 17% in evolutionary). This higher presence of Fuzzy in SC could be related with the origin of the term SC, coined by the father of fuzzy logic, and apparently having a higher recognition in that area. Except for the case of 2007, already explained, the number of papers using the term "Soft Computing" is larger than those using "Computational Intelligence", and the evolution in time does not show any significant difference.

A last comparison of the terms will be performed through a search in "Google scholar". Table 3 shows a comparison of both terms. Each row shows the number of entries (in thousands) containing the mentioned terms (first column), and the terms jointly with SC (second) or CI (third). The first row shows overall entries of SC and CI, while the last one counts on entries on SC and CI without any of the terms chosen to describe the four techniques.

The conclusions from this last table are similar to those previously presented. Similar presence of NN and PR in SC and CI, and lower presence of FL in CI. Around a 10% of entries refering to SC do not mention any of the four associated techniques (close to 20% for CI). The main differences concern the presence of EC that is equaly distributed in SC and CI, the higher presence of PR, and the complete equilibrium of overall entries for SC and CI.

Table 3. Entries in Scholar (in thousands) containing terms related to SC and CI.

| | | SC | CI |
|---|---|---|---|
| | | 63.9 | 63.6 |
| "Fuzzy Logic" | 214 | 17.3 | 8.9 |
| fuzzy | 1980 | 33.6 | 19.2 |
| "Neural Networks" | 588 | 21.4 | 20.9 |
| neural or neuro or neuronal | 1900 | 27.9 | 26.6 |
| "Evolutionary Computation" | 74 | 7 | 8.7 |
| evolutionary or genetic or evolving or evolution | 3610 | 23.8 | 25.5 |
| "Probabilistic Reasoning" | 23.6 | 1.3 | 1.5 |
| probabilistic or probability or bayes or bayesian | 1960 | 20.4 | 22.2 |
| None of the previous terms | | 7.2 | 12.3 |

According to this table, the only difference in between SC and CI is the larger presence of FL in SC, and the larger absence of the four techniques in entries concerning CI.

## 8. Conclusions

Soft Computing could be defined as a computational approach to solve problems under circumstances of uncertainty and/or imprecision, being those circumstances either inherent to the problem (the available information is imprecise) or "added" as a way to cope with (to soften) complexity. It is quite important to realize that the presence of uncertainty or imprecision is not a target for SC. It derives from properties or circumstances of the problem under consideration, or is added as a need to overcome the complexity of the problem.

The core techniques of SC, ordered according to its presence in publications, are Fuzzy Logic, Neural Networks, Evolutionary Computation and Probabilistic Reasoning. Nevertheless, the definition of SC as the mixture of several techniques produces as a consequence a certain absence of clear and fixed borders. This situation generates some criticism and even some controversies when considering if a certain technique is or is not part of SC, but it is obvious that some other techniques should be added to the list. In fact, according to the different definitions of SC that has been considered in the paper, the list should be extended by adding several other components as granular computing, bio-inspired search and optimization, or computing with words.

With independence to the techniques considered as part of SC, hybridization is surely one of the central aspects of the field.

SC is particularly focused on real–world problems for which, quite commonly, the use of a precise and explicit model of the system under consideration, as an intermediate step in its resolution, is either impossible (due to its absence) or infeasible (due to its complexity). Under those circumstances, the strength of SC relies on its ability to work on the basis of approximate and implicit models achieving good (but not optimal) solutions.

It is not easy to find (either from definitions or use) any significant difference among the characteristics, properties and components of Soft Computing and those of Computational Intelligence. So, we can say that both terms apparently refer to the same, or at least a quite related, concept.

## References

1. Zadeh, L.A. Foreword. *Applied Soft Computing*, vol. 1, num. 1, pp. 1–2. 2001.
2. Zadeh, L.A. Soft computing and fuzzy logic. *IEEE Software*, vol. 11, num. 6, pp 48-56. 1994.
3. Bonissone, P.P.; Chen, Y.-T.; Goebel, K.; Khedkar, P.S. Hybrid Soft Computing Systems: Industrial and Commercial Applications *Proceedings of IEEE*, vol. 87, num 9, pp 1641–1667. 1999.
4. Cordon, O.; Herrera, F.; Hoffmann, F.; Magdalena, L. *GENETIC FUZZY SYSTEMS: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*, Advances in Fuzzy Systems - Applications and Theory, Vol. 19, World Scientific. 2001.
5. Li, X.; Ruan, D.; van derWal, A.J. Discussion on soft computing at FLINS96. *Internat. J. Intelligent Systems*, vol. 13, num. 2-3, pp. 287-300. 1998.
6. Bargiela, A.; Pedrycz, W. *GRANULAR COMPUTING: An Introduction*, Kluwer Academic Publishers. 2003.
7. Kercel, S.W. Guest Editorial Special Issue: Industrial Applications of Soft Computing *IEEE Trans. on Systems, Man and Cybernetics-Part C*, vol. 36, num. 4, pp 450–452. 2006.

8. Verdegay, J.L.; Yager, R.R.; Bonissone, P.P.; On heuristics as a fundamental constituent of soft computing *Fuzzy Sets and Systems*, vol. 159, pp. 846-855. 2008.
9. Bonissone, P. Soft computing: the convergence of emerging reasoning technologies *Soft Computing*, vol. 1, num. 1, pp 6–18. 1997.
10. Zadeh, L.A. From computing with numbers to computing with words – From manipulation of measurements to manipulation of perceptions. *Page 1 Int. J. Appl. Math. Comput. Sci.*, vol. 12, num. 3, pp. 307–324. 2002.
11. Pedrycz, W. *Computational Intelligence: An Introduction*, CRC Press. 1998.
12. Duch, W. What is Computational Intelligence and Where Is It Going? *Studies in Computational Intelligence*, vol. 63, pp 1–13. 2007.