Foundations for the Design of a Creative System Based on the Analysis of the Main Techniques that Stimulate Human Creativity

L. De Garrido[1], J. J. Gómez Sanz[2], J. Pavón[2] *

[1] Architecture Department, and Psychobiology Department. Universitat de València (Spain)
[2] Institute of Knowledge Technology. Universidad Complutense de Madrid (Spain)

Abstract

This work presents the design of a computational system with creative capacity, based on the synthesis of the main methods that stimulate human creativity. When analyzing each method, a set of characteristics that the computer system must have in order to emulate a creative capacity has been suggested. In this way, by integrating all the suggestions in a structured way, it is possible to design the general architecture and functioning strategy of a computer system that has the incremental creative capacity of well-known creative methods. This computational system is designed as a multi-agent system, made up of two groups of agents, the *problem solving group* and the *creative group*, the first one exploring and evaluating paths for suitable solutions, the second implementing creative methods to generate new paths that are provided to the first group.

I. Introduction

THE present work aims to explain the basis for a creative computational system based on the synthesis of well-known effective methods that promote human creative capacity.

In a previous work, the implementation of a multi-agent system (MAS) has been shown, capable of emulating the creative capacity of a brainstorming method [1]. Based on this experience, an additional step has been taken to design a computational model that integrates several methods that stimulate the human creative capacity. These methods have been analyzed in order to know the reasons why they stimulate creativity, and to identify a set of hints for the implementation of a creative computational system.

One of the first systematic reviews of methods for producing artificial creativity [2] classified them in three types: combination of familiar ideas (combinational creativity), taking a thinking style and tweaking it (exploratory creativity), and changing dimensions of an existing idea (transformational creativity). Boden [3] also distinguishes between the historical creativity (H), which produces new ideas not known to have been reported at all, and psychological/personal creativity (P), which produces new ideas to the person. Creativity can also be seen as a social construction [4]. It cannot be reduced to some formal properties, but the debate can be useful to enumerate conditions under which an external observer is more likely to consider a system as creative. This criterion is similar to one of the stances when discussing what is artificial intelligence (AI), "intelligence is in the eye of the observer" [5], meaning that a system is as intelligent as an external observer considers it to be.

The four Ps [6] approach is cited frequently to analyze creativity research relating to humans. These four Ps stand for Person (what makes the agent a creative one), Process (what actions need to be undertaken to be creative), Product (what kind of creation is expected), and Press (what cultural context is applied to determine something as creative or not). Computational creativity can be addressed from these four Ps for evaluation, but existing research does not always consider them. For instance, 75% of the papers in the 2014 International Conference of Computational Creativity did not make any reference to social or interactive aspects of creativity, and were more focused on Product and Process aspects [6] [7].

A review on the literature about evaluation of creativity shows arising debates about the definition of terms, lack of autonomy in existing systems, the cultural specificity of many judgments, and the potentially domain-specific nature of creativity [8]. While operational tests (e.g. statistical analysis of the product) have been used for this purpose, questionnaires are a more frequent evaluation tool. Following recommendations from [8], one can address each P with some rules of thumb that include criteria to choose each P depending on the kind of impact one wants to achieve.

Another theoretical framework [9], intends to characterize the different creative systems and concepts, such as uninspiration (failing to be creative in a valued way) and aberration (deviation from the norms) in order to support formal reasoning about creativity, and uses these two examples as illustrative ones. The work presented in this paper is more extensive in the account of techniques. As such, it is more useful as guidelines for those willing to get acquaintance

* Corresponding author.

E-mail addresses: info@luisdegarrido.com (L. de Garrido), jjgomez@ucm.es (J. J. Gómez Sanz), jpavon@ucm.es (J. Pavón).

for creativity techniques. Also, rather than formalizing, it aims to serve as inspiration for enhancing other works. Formal models are more precise, but they are harder to apply than the generic guidelines shown in this paper.

The main contribution of this paper is the definition of an architectural framework, as a MAS, to explore how each method contributes to the generation of creative solutions in an integrated way. This is achieved by first identifying some suggestions from the analysis of each method, which are used as the basis for the implementation of the creative computational system. Their integration is made by defining an architecture of a MAS, which is structured in two groups of agents, the "problem solving group" and the "creative group". This organization is inspired from the functioning of the brain, where the problem solving group would correspond to the executive control network (ECN), and the creative group to the working of the default mode network (DMN). The basic idea is that the first explores and evaluates a set of paths to find a solution. When these are not successful, control is given to the creative group, which will generate new paths for the problem solving group.

First of all, all known systems that stimulate human creativity have been compiled. Many of these systems are variations of others, so the most representative and effective have been chosen.

The following creative methods are analyzed and applied in the design of the agents:

- *Establish analogies with known problems*
- *Creativity matrix*
- *Problem solving*
- *Brainstorming*
- *Variants of Brainstorming*
- *Graphic Brainstorming*
- *www-Brainstorming*
- *Lateral thinking*
- *Parallel thinking*

Sections II to IX describe the analysis of these methods in order to identify elements and functionality for the implementation of the creative computational system. These pieces are organized and integrated as a MAS, which is presented afterwards in section X, as well as considerations for the representation of the information that is managed by the agents. This model is discussed in section XI, taking into consideration other relevant works in the area of computational creativity. The paper concludes with a discussion (section XII) and some final remarks in section XIII.

II. Guidelines for the Design of a Creative Computational System, Based on the Analysis of the Establishment of Analogies

The most common way to solve problems creatively is to identify analogies with other problems that have been previously solved [10].

From birth, human beings begin to interact with their environment, in order to learn to function in it. The first learning tool they develop is imitation and establishing analogies. Human beings begin to imitate other humans in their close environment in order to join the group as soon as possible. This basic mechanism has been enhanced by the human evolutionary system to such an extent that a large number of "mirror neurons" have been created in our brain, whose task is to imitate the activity of those around us. This basic mechanism helps us to survive, integrating ourselves into the group, imitating patterns of action that have apparently been successful for others. Therefore, imitating the actions of others allows us to solve problems in the same way that others have solved. This mechanism can therefore seem very uncreative; however, when applied at various levels of abstraction the results of applying analogies can be surprising and enormously creative.

Human beings, before creating new things, must begin to know the existing ones. The imitation process is very important and occurs just when the human brain is developing, especially in the first four years of our life. Therefore, the imitation mechanism is actually a mechanism that limits human creativity at an early stage and with a low level of abstraction. However, with the passage of time, humans begin to accumulate more and more experiences, and many of them are completely new and unknown to them, and also, due to the very fact of being unknown, they have not been able to learn by imitating the acts of other humans when faced with such novel events. However, humans are able to draw basic parallels between new facts and known facts, in the hope that previously acquired cognitive strategies may help.

It is clear that several types of analogies can be established, with different levels of abstraction. Some analogies have a low level of abstraction and are therefore very close (for example, the analogy between driving a car and driving a truck). Others have a medium level of abstraction and are not so close (for example, the analogy between the activity of a coroner and the activity of a police officer). On the other hand, other analogies can have a high level of abstraction, and can seem very distant, and even belong to different fields of knowledge (for example, the analogy between roasting a piece of meat so that it is tasty and digestible, and superficially burning a beam wood to protect it against fire).

In order to establish analogies in different degrees of abstraction, an adequate representation of knowledge must be previously made, in such a way that the same object must have a huge number of possible attributes, some of which may seem obvious or very general. These attributes must be classified at different levels of abstraction, and they must be able to be activated and deactivated depending on the level of reasoning desired when establishing analogies.

In addition, a learning system must be available. This system will enrich the possibilities, and in the same way, the ability to establish analogies.

As an example to illustrate this mechanism, consider the act of "grilling" food. Hundreds of descriptions can be given to the act of "grilling", and the more descriptions, the more possibilities to establish analogies with similar actions, or in different settings. Grilling can be defined as the act of exposing a food to contact with a very hot radiant surface (such as a frying pan or a griddle), or simply to thermal radiation of a certain intensity, in order to make it more digestible, or with a more attractive flavor for the palate. By defining the act of grilling in this way, analogies can be drawn with any food, as grilling will make it more digestible and tastier. In this way, humans have learned to roast any type of food, or even food leftovers, in order to make them more digestible and tasty. In this way a fish, meat, vegetable, etc. could be roasted.

In this way, human beings have learned to grill any food, and have found that grilling improves its digestion and flavor. However, the concept of grilling is much broader, and has more connotations, and therefore greater applicability.

Alternatively, the act of "grilling" could be defined as the act of exposing food (or any other organic element) to contact with a very hot radiant surface (such as a frying pan or griddle), or simply to thermal radiation of a certain intensity, in order to raise the temperature of the surface of a food to change its physical structure, and make it more resistant to thermal radiation, and thus protect the inside of the food from thermal action. In this way, when roasting a piece of meat its surface is altered, it becomes carbonated, and in doing

so the interior of the meat is protected from the thermal action. As a consequence, the carbonated surface of the piece of meat is stiffer, more crispy and attractive to the palate, and its interior is protected from thermal action, thus preserving its nutritional structure. In other words, when roasting a piece of meat, its outer carbonated structure protects the inside, keeping its nutrients intact, and also the result is more digestible and tasty.

In this way humans have established direct parallels, and have learned to grill any type of food. But they have also learned to establish indirect parallels, and they have learned to roast any other type of organic elements, in order to alter their external structure to preserve their internal structure. For example, we know that if the surface portion of a wooden beam is burned, the wood burned around the edges hardens, helping to protect the interior of the wooden beam from fire. Therefore, learning to grill, we can learn to protect a beam against fire, creating analogies with a high level of abstraction. In fact, this action is called "heat-treated wood", and consists of oversizing the section of wood so that when there is a fire, only the perimeter bark is altered and this scorched bark protects the interior of the beam from the flames of the fire.

Basically this is what case based reasoning (CBR) systems do, although they should be adapted to work with various levels of abstraction. The algorithms that are used in a CBR system for retrieval of past cases, and their reuse and adaptation to the new problem are appropriate for the implementation of the establishment of analogies, which can take advantage of past experiences.

III. Guidelines for the Design of a Creative Computational System, Based on the Analysis of Creativity Matrices

The *creativity matrix* is a method of assured effectiveness to stimulate creativity in solving problems that have a limited range of possible solutions [11][12]. The method consists of a combinatorial explosion between the different possibilities of each aspect that we want to take into account in solving a certain problem. For example, a lighter can be designed with only two attributes in mind: the fuel and the ignition system. By testing all possible values of each attribute and combining them with each other, we can get new and unexpected ideas.

From the analysis of the creativity matrices, it can be deduced that an important component of creative problem solving is the exploration of new search paths, never traveled before. These search paths are based on the association of certain characteristics of the fundamental parameters of a certain problem.

A first computational approach would consist in creating all possible combinations of all the relevant parameters of the objects to be designed. If the number of parameters is under a computational system, it could go through all its possible combinations and associations in a short moment of time, and choose the most appropriate combination, capable of providing the most creative solution. However, if the number of relevant parameters is very high, the possible combinations would grow explosively, the system would be very slow, and it would exhaust the patience of the possible users of the system. That is to say, what is usually called a combinatorial explosion would be created.

To avoid the combinatorial explosion, the different parameters must be evaluated in several different ways, in order to reduce the possible combinations, and make the use of the resulting computational system viable in the application of this system. The restrictions can be the following:

- Removing unwanted combinations of some parameters.
- Enhancing certain desired combinations.
- Assessing each parameter, to encourage, more or less, its use.

- Assessing combinations of combinations.
- Rating each parameter based on the preferences of each user.

In this way the possible combinations between parameters are considerably reduced, and therefore the method would be viable.

As a consequence, a first computational approach to this system would be the Heuristic Search and Solution Tree Pruning systems, although with certain important nuances.

Heuristic search in artificial intelligence is a technique for solving problems whose solution consists of a series of steps that frequently must be determined by systematic testing of alternatives. Therefore, it can be said that heuristic search algorithms are a computational method to solve path-finding problems, that is, "search for the best route from point A to point B" (see Fig. 1).
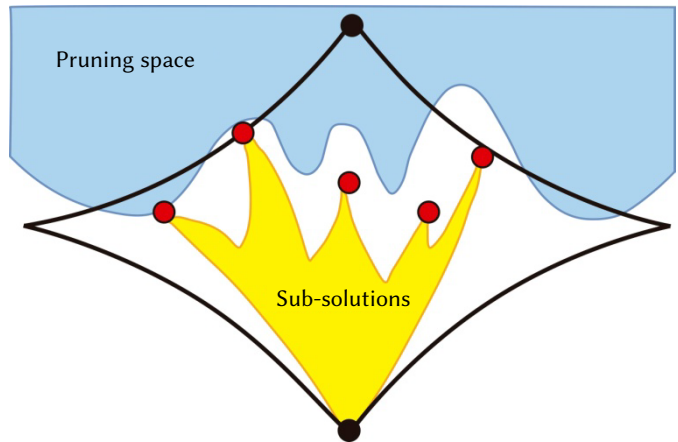

Fig. 1. Delimitation of conventional search areas using tree pruning of possible solutions.

Subsequently, the existence of an e*valuation function* that should measure the estimated distance to the target is assumed. This evaluation function is used to guide the process by selecting the most promising status or operations at each moment. This system does not always guarantee to find a solution, and if it is found it does not guarantee that it is the best.

Heuristic search methods have some information about the proximity of each state to a target state, allowing the most promising paths to be explored first. However, the conceptual functioning of a creativity matrix differs in some determining concepts. One of them is pure chance. Since each tour of the creativity matrix, presupposes that there are always valid solutions, depending on the specific combination of specific parameters that have been initially chosen. In other words, initially a specific combination of parameters is chosen by pure chance, or by a specific personal preference, or out of curiosity, and then a search process is initiated. We should keep in mind that exploring random paths, in the search for possible creative solutions, is a basic and recurring component of any method of stimulating creativity.

On the other hand, chance is also an essential component of the functioning of the human cognitive system. In most cases we are not aware that the brain explores more paths than we consciously explore, and many of them are explored at random (stimulated by concrete experiences during the creative process). In fact, when the brain does not have external stimuli that induce action and decision making, it simply has a default activity, in which it fiddles with possible scenarios determined many times by pure chance.

Therefore, from the analysis of the creativity matrices, two fundamental guidelines emerge in order to design a conceptual model of a creative computational system:

- The "*problem solving group*" must incorporate heuristic search mechanisms to avoid combinatorial explosion.
- The "*creative group*" must incorporate a module for generating random associations. These random associations can be of several types:
  - Absolute chance.
  - Chance limited by similarity (looking for combinations of components that have certain specific attributes in common).
  - Chance limited by strangeness (looking for combinations of components that apparently have nothing in common).
  - Chance limited by personal preferences (looking for combinations of components with certain attributes).

IV. Guidelines for the Design of a Creative Computational System, Based on the Analysis of Problem Solving Strategy (Vertical Thinking)

The problem-solving method is a purely deductive procedure based on cause-effect chains [13]. This thinking can be top-down, or bottom-up. In the first case, a problem is divided into sub-problems and an attempt is made to solve each sub-problem, often dividing it into more sub-problems, until finally all the partial solutions are concatenated, obtaining a possible final solution. In the second case, simple sub-problems are solved, joining with other sub-problems, until a certain situation is obtained that may coincide with an initial statement of the problem.

Vertical thinking, based on a problem solving structure, implies going through a certain path within the tree of possible solutions, from the stride to a certain leaf. Therefore, there is a huge number of possible paths within the decision tree until a possible decision is reached (see Fig. 2 and Fig. 3). This is why strategies must be established based on a pruning of the decision tree, choosing priorities to travel as a priority only certain paths until reaching a possible solution [14]. It is the most widely used approach in computer systems; in fact, it has a certain coincidence with the operation of rule-based systems.
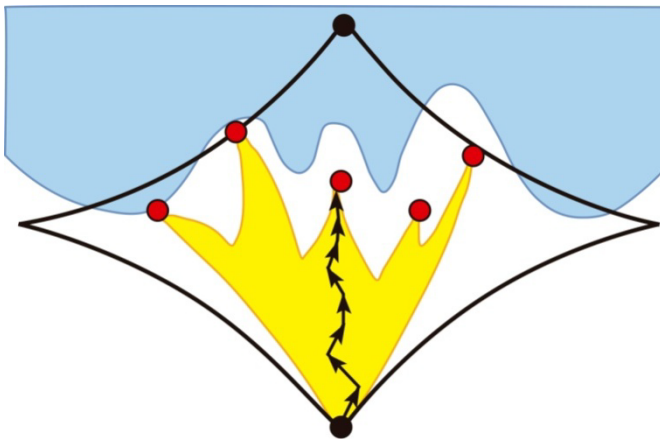

Fig. 2. Ideal and successful path to find solutions through conventional problem-solving systems.

However, the problem with rule-based systems is that they produce a multitude of solutions when the problem is complex and poorly defined. On the other hand, if the field of action is reduced so that it is well defined, the usual solutions are not creative at all.
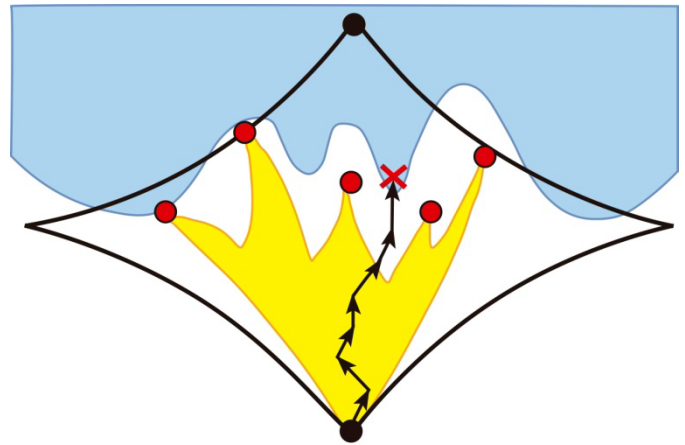

Fig. 3. Possible and unsuccessful paths to find solutions through conventional problem-solving systems.

In some cases, when the algorithms are very complex, the system could deduce some solution that might seem surprising or creative, simply due to the complex algorithmic interactions. However, creative solutions would be a rarity, since the system is based on the logical concatenation of the most appropriate actions in each case, which is why they tend to leave out a huge number of novel, surprising and therefore creative solutions.

For this reason, these problem solving systems must be complemented with other systems, which will be activated when it is not able to find novel solutions. In other words, systems based on the problem solving structure could come up with some creative solutions on their own, but in the event that it fails to generate sufficiently creative solutions, or simply generates an insufficient number, the system must give up control to other alternative systems.

The most important suggestion here is that the agents of the "*problem solving group*" should be implemented in a conventional way, following some rule-based technique, for instance. If the system can find a solution, but if it is not considered creative enough, this group cedes control to the "*creative group*", which has other more creative complementary structures.

V. Guidelines for the Design of a Creative Computational System, Based on the Analysis of Brainstorming

Brainstorming is probably the oldest and best known creative problem solving technique [15][16]. The main objective of *Brainstorming* is to break the usual limitations of the human cognitive system, and generate a large number of ideas, many of which can be really creative and solve a certain problem.

The structure of Brainstorming suggests that a MAS is a suitable computational model of a group of individuals that collaborate to find a solution. In this system, each agent does not have to be specialized in a specific task, but instead each agent must have a different structure and therefore different behavior. In other words, each agent must have a different algorithmic structure than the others, and therefore must process information differently. All agents share the information in a common space, in such a way that the information generated by a certain agent can be useful to another agent, who uses it and generates new information in the common space.

Based on the information generated at each moment, each agent can decide to work in a linear way, analyzing the information and deducing new information, or randomly, to a greater or lesser degree.

To design the basic architecture of the MAS, the functional structure of a Brainstorming must be analyzed, which is based on four fundamental rules.

A. Eliminate Criticism

The elimination of criticism in the idea generation process is an important factor to take into account when designing a creative computer system. This factor clearly indicates that to promote creative thinking, each possible solution should not be evaluated as soon as it is generated, since, even if it is a bad solution, it could serve as starting information for another agent who may be able to propose a creative solution. That is, ideas should not be eliminated early as they arise, since even if a given idea is not valid, it can stimulate any of the other agents to generate new ideas, which could be very creative.

B. Absolute Freedom

Absolute Freedom suggests several things.

- Each agent may have a different structure.

- The generation of ideas, although they may seem absurd, should be stimulated, since during the process, they could stimulate the generation of truly creative ideas that solve the problem. At any intermediate stage of the process, it might seem that the solutions that are being generated will not be adequate (if they were evaluated at the same time), but as the process continues, there could be unexpected changes and complex feedback that could lead to one or more valid unexpected solutions, new and therefore creative. For this reason, there should be some agent in charge of generating ideas randomly ("random agent").

- The ideas generated must be stored in a certain file, in order to be evaluated in a second stage.

C. Generate a Lot of Ideas

The computer system should create as many ideas as possible, in the common space of the MAS, in order to have a better chance that some of them can be successful. In addition, in the evaluation stage, the degree of creativity of each solution can be assessed, and in this way the system can learn, in order to explore paths close to those by which ideas have been generated considered as more creative. Therefore, the system should have a *learning system* that generates information for a certain agent, so that their subsequent proposals are based on previous successes.

D. Multiplying Effect

The ideas proposed by each agent throughout the process may have been generated as an extension of the ideas that they have previously proposed, but especially they must be provoked by the ideas proposed by the rest of the agents. That is, each agent develops its activity as a consequence of the activity previously developed by other agents. In this way, the ideas generated by each agent feed the rest of the agents, generating a multiplier effect of the ideas, which vary subtly, depending on the essential parameters of each agent.

It would therefore be advisable to have some agent generates ideas contrary to those generated by the other agents (which is not an evaluation or criticism of the rest of the ideas, it is simply a new idea contrary to those that are being proposed). This agent could be called "*tenth agent*" (the "tenth agent can be the same agent as the "random agent", but assuming a different role, therefore, from now on it will be called "*spark agent*"). Therefore, and as a result of this collaboration of agents, the system may be able to solve the problem in surprising ways.

Once the basic parameters of a MAS based on the creative structure of a *Brainstorming* have been established, it is convenient to analyze the development of a work session, and therefore determine the roles that each agent should have based on their operation.

1. Secretary

An agent must take care of the logistics of operation of the group of agents, organizing the order of intervention of each one, as well as their priorities ("*secretary agent*"). In the same way, it must be in charge of storing all the ideas generated by the set of agents.

2. Relaxed and Cheerful Atmosphere

Today it is known that a relaxed and joyful environment induces the brains of well-trained specialists to switch modes of operation, to deactivate the ECN and activate the DMN [17]. A relaxed environment induces the brain to function in a "*mind wandering*" way, and problem solving is taken to an unconscious plane, in which automatic and spontaneous cognitive mechanisms take place. When this mode of operation is induced, the brain manipulates the information, adding, eliminating and transforming existing information, mixing it with random information, unrelated to the problem to be solved.

At a computational level this means that each agent in the group must have a different internal structure, and therefore they can manipulate the information in a different way. Alternatively, each agent can assume different roles, and therefore manipulate the information in a different way. In any case, agents must be able to remove or add information randomly, mixing it with the information that has been generated, allowing them to explore unexpected paths. In this way, novel and unexpected solutions can be obtained.

Therefore, if after a certain period of time, the MAS has not been able to solve a certain problem creatively, it can change its operating mode, assuming different roles and generating random information, mixing it with all the information that is it has accumulated throughout the process.

3. Short Duration

The long working sessions generate weariness in the participants and cause the participants to repeat over and over again established ideas. The reason is due to the fact that once a certain set of ideas has been generated, the participants tend to focus on their development rather than on the generation of new ones. Once certain search paths have been explored, certain neural connections are subtly reinforced, inducing the search paths associated with those connections to be explored again and again. For this reason it is preferable to do short sessions, and to continue new sessions after a certain time (for example after one or two days).

This way of functioning of the brain has been shaped throughout human evolution, since it guarantees our survival. The fact that humans continue to make the same conventional decisions that previously have been proven safe, has an evolutionary advantage over making continuous changes, and exploring new, unpredictable and less safe paths. Only when a strong conflict occurs, the brain stimulates the abandonment of certain patterns of activity and the adoption of new patterns, exploring new ideas.

Computer systems do not have these limitations so this aspect is not relevant for the design of a computer-based creative system.

VI. Guidelines for the Design of a Creative Computational System, Based on the Analysis of Brainstorming Variants

Although with an identical basic structure, many variations of Brainstorming methods have been proposed in order to increase their creative efficiency. Therefore, when analyzing each variant, certain complementary guidelines can be established, in order to optimize the structure of the MAS.

In general, the computational guidelines derived from the analysis of the different types of *Brainstorming* do not suggest the modification of the architecture of the MAS, and only suggest alternatives in its functioning structure, and the adoption of different roles by the different agents. Therefore, the "secretary agent" could make sure that the group of agents works in one way or another, according to the specifications shown in the different varieties of *Brainstorming*.

A. Stop-and-go Brainstorming

This method suggests that consciously delving into the problem moves in a certain direction, and the creative scene shifts to a new environment. In this new environment, ideas are generated again based on the initial approach and based on the information deduced by each participant of the group, which moves the scene to a new unpredictable creative environment, which contains the different perceptions of each component of the group.

This fact suggests that the computational system must alternate two types of thinking. One type of thinking is more spontaneous and random, and another type is more deductive and linear (problem solving).

B. Sequential Brainstorming

This method suggests that each participant goes deeper into a certain idea, since he accumulates all the information exposed by the others.

From a computational point of view the architecture of the system is the same and only the mode of operation varies. The different agents must be activated in an ordered sequential manner based on the information generated at each moment. In this way, an ordered list must be made with the order of activity of the different agents, in such a way that each agent will only be activated when the activity of the previous agent has finished. The work session will involve several cycles of activity of the different agents, and in each cycle the order of action of the different agents must be different. The "*secretary agent*" is in charge of deciding whether the MAS works in sequential mode, or in random mode.

C. Constructive-destructive Brainstorming

This method is very interesting and its effectiveness lies in stimulating creative ideas in an environment of reduced possibilities. The fact of generating destructive ideas in a first stage eliminates a priori certain conventional and easy search paths, in order to focus on the generation of ideas through other alternative and opposite paths.

From a computational point of view, this method can be carried out by initially changing, in the first operating cycles, the roles of the different agents, in order that they only generate destructive ideas, so that in a second place (in the following cycles of operation), change roles again, and engage in generating constructive ideas.

The "*secretary*" agent is in charge of deciding whether the MAS works in a *destructive mode*, or in a *constructive mode*.

D. Individual Brainstorming

This method does not suppose additional information when modeling a creativity computational system.

E. Anonymous Brainstorming

This method does not suppose additional information when modeling a creativity computational system. The only difference is that the way of acting of the agents is not conditioned by the proposals and behavior of other agents.

F. Brainstorming with Post-it (TM)

This method does not suppose additional information when

modeling a creativity computational system. In a computer system, the activity of each agent does not have to be conditioned by other more prestigious agents (although in a conventional *Brainstorming* it can be programmed that some agents have a certain priority over others if that were the case). This activity can be done by the secretary at any time it is deemed necessary, since the secretary has information on the effectiveness of each agent and can provide them with different priorities.

Therefore, in a MAS architecture the different agents involved may have a similar weight, or on the other hand, in certain operating cycles the importance of the activity of certain agents can be weighted over others, in such a way that the information they generate is a priority (for the rest of the agents in subsequent operating cycles) over that generated by the other agents.

G. Brainstorming Phillips 66

This method does not suppose additional suggestions when modeling a creativity computational system.

H. Brainstorming Buzz

This method does not suppose additional suggestions when modeling a creativity computational system.

I. Didactic Brainstorming

This method is very interesting when modeling a creativity computational system since it seems to suggest that "*fuzzy*" searches should be created in parallel that are generally valid and none of these should be specified too soon. The information should be managed gradually and obtain general deductions without specifying. This aspect suggests that the "*creative group*" should work as a whole with various levels of abstraction, as it did with the "*problem solving group*". Each agent can act in parallel with various levels of abstraction, depending on the user's specifications, controlled, again, by the "*secretary agent*".

J. Brainstorming SIL (Successive Integration of Solutions)

This method is very suggestive since it proposes paths to follow in the process of developing ideas based on the forced grouping of previous ideas in an incremental way.

K. Brainstorming 635

This method is even more effective than the previous method, since it forces each participant to break the usual cognitive path several times, forcing them to explore lesser-known paths. The computational structure can be similar to that based on the previous method.

L. Brain Writing

This method does not suppose additional information when modeling a creativity computational system.

M. Collective Notebook

The fact of waiting one day between the generation of ideas sessions is based on the partial forgetting of the idea generated previously, so that the new idea arises from a collateral and non-evolutionary state. That is, the method aims to prevent it from deepening into a line of thought and instead tracked laterally. Therefore, it is an alternative to *Didactic Brainstorming*, and the same comments are valid to guide the development of a computational system.

N. Brainwriting Pool

The method is a new way of sharing previous ideas, although the computational structure may be similar to that of the previous methods.

O. Delphi Method

This method provides feedback to deduce new ideas. It is interesting and it is basic for a creativity computational system based on agents that interact with each other, feeding back a situation.

P. Nominal Group Method

This method is crucial, it is very effective since it generates many ideas and ensures that they are good without the need for a subsequent evaluation stage since the evaluation is continuous. As with the previous method, this method seems to suggest an agent-based system in which the group analyzes and advances the idea that each of them has generated to force again each of them to suggest a new idea based in those exposed as an extension of the one suggested previously.

VII. Guidelines for the Design of a Creative Computational System, Based on the Analysis of Graphic Brainstorming

Graphic Brainstorming [18] intends to facilitate the generation of graphic creations, such as architectural design, logo design, advertising image, fashion design, etc. This type of *Brainstorming* is complex, since the generation of abstract ideas must be complemented with the generation of graphic ideas, in the form of sketches, which represent the germ of the graphic composition that is intended to be achieved. Therefore, *Graphic Brainstorming* consists of at least three stages: idea generation, graphic generation and evaluation. And there are three methods to implement it:

- Shape Brainstorming. This method analyzes how parallels can be made between concepts and certain forms. For example: "designing a house with a rounded shape" would be something easy to specify. It is easy to find analogies between the circular concept and a circular shape to generate an initial action pattern. Instead, "designing a house with lots of light" would be much more difficult to establish initial formal analogies to generate initial sketches that can be refined in a later process. The key without a doubt is that the knowledge of things must be very extensive, and also ambiguous.

- Symbolic Brainstorming. This method is even more effective than the previous one, since the problem lies in finding a suitable symbol, but each symbol has a strong spatial and formal character, which can serve as a starting point for the design process. For example: "designing a very aggressive home". The "aggressive" concept could be associated with a triangular or star shape, for example, or any existing symbol with many edges such as the triangular shape, the star shape, or similar. A computer system should make a great collection of previous associations between concepts and symbols, which means that a certain symbol should have a great quantity of attributes, and with ambiguous value.

- Metaphorical Brainstorming. This method is similar to the previous one, although more abstract, since the problem lies in finding a suitable metaphor, and each metaphor can again be associated with concrete or abstract forms, which can serve as a starting point for the design process. For example: "designing a house that stimulates spirituality". The concept "spiritual" could be associated with common metaphors such as "ascension to heaven", "communion with God", etc. and these to forms usually associated with religions in a certain culture, such as a form of Latin cross, Greek cross, triangles, circles with radii, or a combination of them, for instance. A creativity computer system should make a great collection of previous associations between concepts and metaphors, which means that a certain symbol should have a multitude of attributes, and with ambiguous value.

The three *Graphic Brainstorming* methods suggest that both the "problem solving group" and the "creative group" should work at various levels of retraction, as already mentioned. This implies an important effort when it comes to representing knowledge, and when implementing the different processes for modifying, filtering and adding information.

VIII. Guidelines for the Design of a Creative Computational System, Based on the Analysis of Www-Brainstorming

This method is a grouping of *Shape, Symbolic and Metaphorical Brainstorming*, but much more powerful, since the system incorporates a search engine for forms, symbols and metaphors on the web.

The analysis of the system suggests that the "creative group" of agents must have a specialized agent that looks for formal precedents, symbols, and metaphors in the web (therefore, it is called "www agent"). This search for associations in the web provides feedback to the system, and can be done at any time, although it is especially important at the beginning of the creative process.

This way of adding information to the system in order to stimulate new unexpected search paths is especially important, since the information that is added is not completely random, but has a certain type of connection with the problem to be solved.

IX. Guidelines for the Design of a Creative Computational System, Based on the Analysis of Lateral Thinking

Lateral thinking is a specific way of organizing thought processes, to find a solution through unorthodox strategies or algorithms, which would usually be ignored by logical thinking [19]. The mechanisms that promote *lateral thinking* have a direct influence on the conceptual architecture of the computational system to be designed. The strategies that promote lateral thinking are innumerable, and only the most frequent and effective are considered here. Other possible strategies do not have an impact on the design of the system, and only provide minor suggestions on the role of the different agents involved.

A. Random Words (Random Input)

A computer system does not have the cognitive restrictions of a human brain, so it is not difficult to break a certain logical structure of thought. The problem is deciding which logical structure to program. This technique suggests that a creative system should be based on two complementary and alternative problem-solving structures. First, a structure must be well defined, designed under intrinsic human parameters, and must be capable of solving a certain problem from a conventional point of view, in the way that most humans would, according to previously accepted conventional information and parameters. Another structure must be in charge of randomly breaking different processing stages of the previous system, introducing certain random information, more or less related to the problem to be solved.

Therefore, the analysis of this technique suggests that the creative computational system is based on a group of agents that can change roles when no truly creative ideas have been generated. That is, the group of agents with a "conventional role" would try to solve the problem. In the event that they cannot solve it, they would change their role, and adopt a "creative role" completely modifying their way of operating (in this case they can generate random proposals) and trying to reach a creative solution.

A more interesting alternative, which is emerging through this analysis, is that the MAS is made up of two groups of agents.

The "*problem solving group*" would try to solve the problem under conventional rational parameters that optimizes the obtaining of creative solutions in various ways. In the event that it cannot come up with any truly new solutions, it would cede control to the "creative group", whose agents have a completely different role, and in this case they can even come up with random ideas. In this way, the path for a solution seems wobbling as it is shown in Fig. 4.
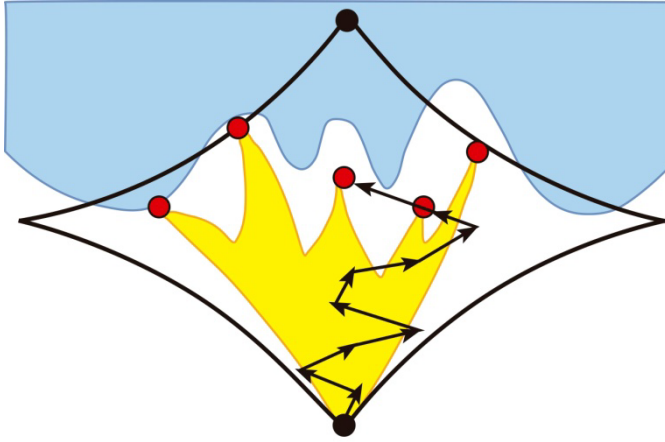

Fig. 4. Unpredictable path that allows one or more creative solutions to be achieved, using lateral thinking.

B. Delete Some Characteristic of the Problem

The analysis of this method suggests that the system must have a component that can manipulate the information within the "*creative group*" of agents. As it is explained in section X, in order to establish analogies at a high level of abstraction, the structure of the object-attribute-value information must have as many attributes as possible, and with the greatest possible variety of values. However, both the range of possible attributes and possible values must be continuously modified in order to find new exploration paths, and therefore obtain new ideas. The same can be said with respect to the characteristics of the problem to be solved. The problem to be solved can be altered, both in its initial definition and in the expanded definition that is completed as the resolution process progresses. This work could be done by an agent specialized in adding, filtering or modifying the information, the "*info-modifying agent*" at all times, under the control of the "*secretary agent*".

C. Modify or Exaggerate any Aspect Related to the Problem

The analysis of this method suggests, as it was the case with the previous method, that the "*creative group*" of agents should incorporate an agent ("*info-modifying agent*") in charge of modifying some attributes of the information related to the problem. This information can be altered, minimizing to exaggerating certain aspects of the information related to the problem. In this way, new paths can be explored, which if the information had not been modified would not have been explored, or simply nearby paths would have been explored.

D. Establish Analogies with Other Situations or Problems

As it has been said in the analysis of the establishment of analogies to solve problems, it is to base the establishment of analogies, at different levels of abstraction. Therefore, the information must initially be structured in several levels of abstraction, so that the "*creative group*" of agents can manipulate it at each different level in order to establish analogies with other previously solved problems.

E. Reverse the Problem

This method introduces distorting elements to avoid the chain of ideas that tends to be created in any conventional problem-solving system. Any algorithm is based, even if it is not desired, on certain chains of ideas, which often go unnoticed when programming. The same learning algorithms in reality what they do is a chain of ideas, it is just the opposite that is intended when implementing a creative system. In fact, the learning algorithms try to solve a certain problem based effectively, using the most appropriate paths in the decision tree, and therefore avoid exploring new paths.

The "*creative group*" should incorporate an agent ("*chunking agent*") that is exclusively dedicated to breaking the chain of ideas that tends to be repeated, and encourages the search for new paths, although initially they seem less promising. This agent must fragment the information usually united, and recompose it in different ways, although they may seem absurd. In this specific case we must reverse the problem. For example, when faced with the problem "designing a ship that rests on the sea", we could propose: "design a sea that rests on a ship". In fact, this idea was what drove the design of the hovercrafts.

F. Break the Problem into Different Components

This method is very useful, and in fact it is a basic strategy of the "*problem solving*" systems, called "divide and conquer". In this way to solve a problem it can be divided into parts, which can be solved separately, and to do so, each part can be divided again into parts that can be solved separately. At the end of the process, each sub-problem is so simple that it has an easy solution, and by concatenating all the solutions, a method is obtained to solve the general problem.

For example, to try to design a home, the kitchen, the bedroom, the living room, etc. can be designed separately, and in the same way, to design a kitchen the bench, the work area, the table, the furniture, etc. Once all the parts have been designed, the design of a house has been achieved.

In this sense, the "*problem solving group*" of agents should initially be able to identify parts of the problem, and once identified they can begin to be solved using a bottom-up structure. The problem with this strategy is that the link between all the parts must be established so that the whole is harmonious and well composed.

G. Take the Problem Out from Its Usual Context

By taking a problem out of its usual context, we are actually breaking pieces of information that are usually perceived as joined, and by doing so the pieces of information can be put back together in a different way.

Therefore, the "*chunking agent*" of the "*creative group*" must be able to establish random groupings of certain attributes of the problem to be solved with different values than the usual ones.

By randomly changing some of the initial conditions, the system would work in a strange and unpredictable way, would explore unsuspected paths and could arrive at some creative solution.

X. Guidelines for the Design of a Creative Computational System, Based on the Analysis of Parallel Thinking

Parallel thinking [20] allows contradictory opinions to coexist in parallel without having to be correct at every step, and in which there is no clash, no dispute, and no "true/false" initial judgment. In other words, this method basically indicates that a solution is good even though it has several adverse aspects.

Therefore, a computational approach would be based on the fact that some objectives delimited a priori could not be fulfilled and

nevertheless take a certain solution as good. In this sense, several solutions could be found that, although they do not meet all the objectives, meet a minimum of essential basic objectives. However, these solutions are so novel that they can be taken for granted. Therefore, the "*secretary agent*" must be able to establish a hierarchy between the specified objectives, grouping them into indispensable, important and secondary. In the same way, the representation and manipulation of knowledge through *fuzzy logic* techniques must be endowed with a certain ambiguity.

XI. Conceptual Model to Implement a Creative Computational System, Based on the Analysis of the Main Methods that Stimulate Human Creativity

The suggestions made in the previous sections establish the principles for the design of the conceptual structure of a computational system capable of emulating the activity of the set of main methods that stimulate human creativity. The next subsections present the architecture for the information processing system and the information representation system.

A. Information Processing System

1. Definition of the Multi-agent Creative System

The collaboration of specialized agents organized as a MAS facilitates the implementation and integration of the methods and techniques suggested in the previous sections. This MAS is structured as two groups of agents: the "*problem solving group*" and the "*creative group*", as shown in Fig. 5.

The "*problem solving group*" has a basic structure containing at least two agents, the "*generator agent*" and the "*evaluator agent*". The problem solving group includes different algorithmic techniques of conventional AI, through which the tree of solutions is explored in search of adequate solutions. The only difference is that it can work at different levels of abstraction, and that is why systems based on the establishment of analogies take on a new creative character. Therefore, the generating agent can adopt several roles, and several levels of abstraction, while the evaluating agent simply limits itself to evaluating the possible solutions (both from the generating agent and the creative group) to accept one or more solutions.
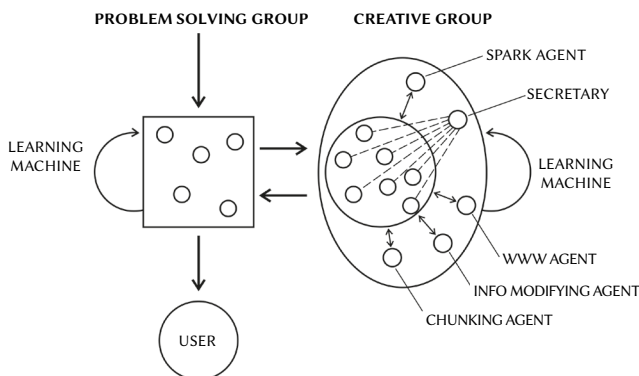

Fig. 5. Organization of a multi-agent creative system based on the analysis of the main methods to stimulate creativity.

The "*creative group*" is made up of a greater number of generic agents (between 5 and 7 agents), together with five specialized agents: the "*secretary*", the "*www agent*", the "*info-modifying agent*", the "*chunking agent*" and the "*spark agent*".

The "*secretary*" controls the operation of the "*creative group*", proposing the order of activation of the rest of the agents (sequential-random), changing their role (destructive-constructive), as well as their priority.

The "*www agent*" accesses the web on a regular basis, in order to find conceptual and formal analogies to the ideas generated by the group of agents in the group.

The "*info-modifying agent*" is in charge at all times of eliminating, adding or modifying the information related to all the possible objects involved in the resolution of the specific problem to be solved.

The "*chunking agent*" is exclusively dedicated to breaking the chain of ideas that tend to be repeated throughout the operation of the "*creative group*".

The "*spark agent*" is responsible for generating random ideas, or ideas contrary to those that may be generated at any time.

Both groups must have a "*learning system*", in order to learn from all the solutions generated, which ones have been finally chosen by the user, and they must know the differential reasons for which they have been chosen. The *learning system* has two components: the "abstract learning system", and the "details learning system". In order to avoid associations of a large number of pieces of knowledge, that is, in order to avoid the "chunking" of information that would force the system to repeat previously valid solutions, and therefore less creative.

The relationship between the different methods to stimulate creativity and the architecture of the system is as follows:

- *Establish analogies with known problems*. It is carried out by the "problem solving group"
- *Creativity matrix*. It is carried out by the "problem solving group"
- *Problem solving*. It is carried out by the "problem solving group"
- *Brainstorming*. It is carried out by the group of agents of the "creative group"
- *Variants of Brainstorming*. It is carried out by the group of agents of the "creative group"
- *Graphic Brainstorming*. It is carried out by the group of agents of the "creative group", in collaboration with the "www agent" and the "chunking agent"
- *www-Brainstorming*. It is carried out by the group of agents of the "creative group" in collaboration with the "www agent"
- *Lateral thinking*. It is done by the "chunking agent", the "info-modifying agent" and the "spark agent"
- *Parallel thinking*. It is done by the "chunking agent", the "info-modifying agent" and the "spark agent"

2. Functioning of the Multi-agent Creative System

In general, when the "*problem solving group*" is active, the "*creative group*" is not active, and vice versa, that is, they usually work in an antagonistic and complementary way.

Initially, the "*problem solving group*" is activated, and it will continue active until it finds a solution to a certain problem. The "*problem solving group*" can work at several levels of abstraction, therefore it can be activated in several sequential cycles, until the solution is properly specified.

If the group does not get a solution, or if the solutions it gets are not satisfactory, the "*problem solving group*" gives the control to the "*creative group*", which will be working until it achieves several solutions to the proposed problem. The list of possible solutions generated by the "*creative group*" is passed back to the "*problem solving group*" to be evaluated. Finally, the ideas that pass the evaluation process are finally presented to the user.

The "*creative group*" has a greater capacity to process information and has a greater number of agents, and can manipulate the information by adding information more or less related to the problem to be solved, and even random information. When adding, removing or modifying information, the rules that could be activated are different, so the troubleshooting strategy can vary considerably from one cycle to another, so the results are unpredictable.

B. Structure of Information Representation

The representation of knowledge is essential when designing a computational system that emulates the creative methods previously analyzed. In principle, the most appropriate representation is through the use of "object-attribute-value" structures for a specific object, idea or thought. This way of representing information has proven to be valid to represent even abstract and complex concepts, such as the existential rhetoric of architectural space [21].

First of all, the information on each object must be as extensive as possible, so it must have as many attributes as possible. In addition, it must have the largest possible number of relationship specifications with other objects, and it must have perfectly defined constraints. Finally, the range of possible values of attributes must be perfectly delimited, but with the greatest possible number of variations.

The greater the number of attributes and the greater the number of possible values for each attribute, the more creative the system can be. Let's take an example. A teacher does not have a ruler to draw a straight line on the board. And it will look at all candidate objects in the classroom that have a "weight" (very low, low, medium), a "length" (medium), and a "shape" (straight). Surely in the classroom there are candidates such as a drawer, a notebook, a tensioned cable, a computer keyboard and also a chair. The chair could have straight legs. In this way the teacher takes the chair and draws with it on the blackboard (see Fig. 6).

Fig. 6. Example of using lateral thinking for creative problem solving.

Students might think that their teacher is crazy (because he does unusual things) but also that he is very creative, or even a genius. The key is in how the knowledge of the objects in the classroom has been represented, and specifically the chair. For the teacher to use it to draw a straight line on the board, the chair must have the attribute "weight" (with a very low, low, and medium range of values) and the attribute "legs" with the attribute "shape" (with a range of values that includes the straight shape), and the attribute "length" (with a range of values that includes the average length).

Therefore, the initial representation of the information must be exhaustive, with the largest possible number of attributes, and with the largest possible range of values (see Fig. 7). In addition, due to the learning system, the information is gradually enriched with the activity of the system, being able to modify the structure of the information, the number of attributes and its range of values.

```
object: chair
    is a:               furniture
    utility:                school, dining room, office, living room, beach
    mobility:               fixed, mobile, articulable, folding, recessed
    weight:                 very light, light, medium, heavy, very heavy
    volumen:                very small, small, medium, large, very large
    resistance:             very low, low, medium, high, very high
    reconfiguration level:  very low, low, medium, high, very high
    requirements  soil, illumination, humidity, local size, environment color, etc.
    relationship with other objects furniture
                                lamps
                                doors
                                windows
                                ground
                                walls
                                ceiling
    components
        legs
            without legs
                hanging from the ceiling
                side braces
                wall anchor
                furniture anchoring
            with legs
                relation to     seat        none, direct, indirect
                                soil        none, direct, indirect
                                arms        none, direct, indirect
                type            fixed, extendable, articulating, folding
                number          none, number
                shape           straight, curved, polyline, polycurve
                length          very short, short, medium, long, very long
                thickness       very thin, thin, medium, thick, very thick
                texture         smooth, rough, silky, etc.
                material        wood, iron, steel, plastic, etc.
                color           white, black, red, blue, etc.
                protection      varnish, chrome, painted, etc.

        seat
            without seat
            with seat
                relation to     legs    none, direct, indirect
                                soil    none, direct, indirect
                                arms    none, direct, indirect
                endorsement none, direct, indirect
                folding type    fixed, extendable, articulating,
                shape           straight, curved, polyline, polycurve
                length          very short, short, medium, long, very long
                thickness       very thin, thin, medium, thick, very thick
                texture         smooth, rough, silky, etc.
                material        wood, iron, steel, plastic, etc.
                color           white, black, red, blue, etc.
                protection      varnish, chrome, painted, etc.

        back
            without backrest
            with backrest
                relation to     seat    none, direct, indirect
                                soil    none, direct, indirect
                                arms    none, direct, indirect
                                legs    none, direct, indirect
                type            fixed, extendable, articulating, folding
                shape           straight, curved, polyline, polycurve
                length          very short, short, medium, long, very long
                thickness       very thin, thin, medium, thick, very thick
                texture         smooth, rough, silky, etc.
                material        wood, iron, steel, plastic, etc.
                color           white, black, red, blue, etc.
                protection      varnish, chrome, painted, etc.

        arms
            without arms
            with arms
                relation to     seat    none, direct, indirect
                                soil    none, direct, indirect
                endorsement none, direct, indirect
                legs            none, direct, indirect
                type            fixed, extendable, articulating, folding
                number          one, two
                shape           straight, curved, polyline, polycurve
                length          very short, short, medium, long, very long
                thickness       very thin, thin, medium, thick, very thick
                texture         smooth, rough, silky, etc.
                material        wood, iron, steel, plastic, etc.
                color           white, black, red, blue, etc.
                protection      varnish, chrome, painted, etc.
```

Fig. 7. Example of dynamic and exhaustive representation of information that allows the representation of information with various levels of abstraction, and the subsequent manipulation of the same by the "creative group" of agents. As a result, different representations (alternative and complementary) of the same object will coexist at each stage of the design process.

This general information is always available in the database, although each time the system tries to solve a certain problem it handles it differently. First, the system ("info modifying agent") classifies the attributes of the information based on their priority for

a specific problem. Second, the system can filter the information in various ways. Third, the system can manipulate the information by adding information semantically related to the problem to be solved. Finally, the system can even add random information (see Fig. 8).
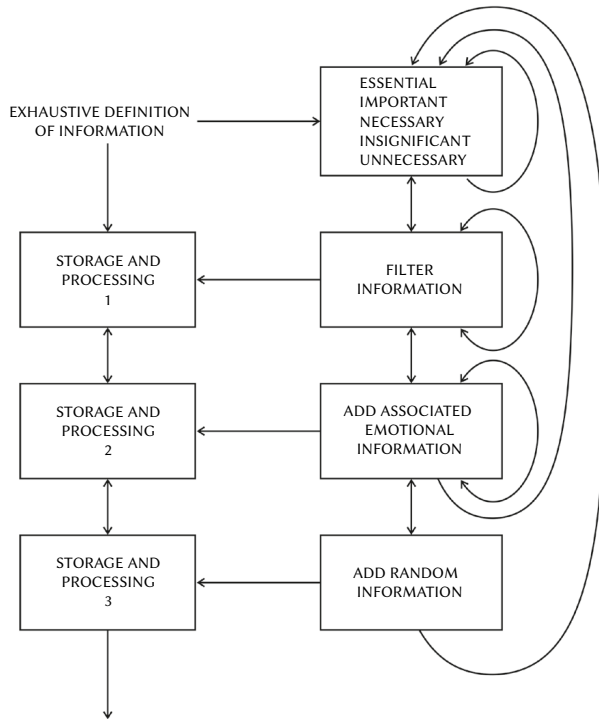

Fig. 8. Information processing system for the "creative group" of agents. The initial representation must be comprehensive and must be dynamically enriched through the learning system. However, the information can be modified continuously at each stage of the creative process, creating different, alternative and complementary representations of the same object.

The manipulated information can be temporarily stored during the resolution of a certain problem, but this does not alter the initial information available in the database at all times. Therefore the system has at all times several representations of the same object (called "virtual" representations).

In this way the agents can dynamically filter the information, making it "virtual", since at any moment the original structure of all the objects involved in the process of solving a certain problem can be recovered. In this way, the information available from the outside world is increasingly complex, with which the range of alternatives is increasing, and therefore the creative component of the system.

A key aspect is that the different attributes of objects must be organized, grouped and classified according to their importance. In this way, the dynamic and blurred manipulation of the information can be allowed, grouping the attributes in different ways or changing their priority at all times. This mechanism would allow the codification of the information storage and its dynamic and fuzzy retrieval, that is, the information, although apparently the same, would be reconstructed differently in each specific case. In this way, the fuzzy and dynamic mechanisms of human memory would be emulated.

XII. Discussion

The guidelines obtained in this paper can contribute to enhance existing works on the application of MAS for building creative systems. Some of the most relevant are reviewed below, with some indications on how our framework integrates and expands those ideas.

López-Ortega [22] considers deliberate creativity (the result of processing existing information), spontaneous creativity and human in the loop creativity (assisting a human operator in the creation act), and how agents can achieve them using planning, divergent exploration, selective attention, combination, and resolving. Creativity would result from the combination of these processes. Our work, however, goes further in the identification of processes and how they contribute to achieving creativity.

A model of organizational creativity [23] (i.e., the one organizations promote to survive producing new ideas, products, or services) proposes the collaboration of monitoring agents (checking the opinion of other agents), capturing agents (retrieving opinions from a database), and creative agents (retrieving ideas from a storage). Compared to our work, they address a limited number of creativity techniques, focusing in the identification of categories to evaluate the solutions.

Something in that line is the proposal by Macedo & Cardoso [24], who conceive the creation as a combination of three agents: the author agent (that produces the surprising products) and the jury-agents (that decide how surprising they are and can update the author agent emotional state towards its creation). They define the creation as a try-and-error process guided by utility functions, and the appraisal method. This author-agent and jury-agent scheme would be implicit in ideas of our framework, such as "eliminate criticism" (section V.A) to capture what the author agent does when the jury-agent does not show as much surprise as expected.

The SMART formal agent framework has been used to analyse creativity [25]. Creativity is seen as the result of autonomous interaction between parties and could be identified in different inherent activities to an agent framework, for instance in the perception of the agents, in the revision of models when taking decisions. Our approach considers the reverse, how creativity can be explicitly realised by agents by using human metaphors. These metaphors are important to understand what happens within the MAS and what the intended results are. After all, it is the observer who judges the existence of creativity.

Agents have been applied in the conceptualization of creativity in painting, with agents controlling fitness functions of ant-colony optimization software algorithms [26]. The work applies rather straightforward modifications of algorithms, letting the reader without an explicit guidance of why that particular approach makes sense. In this case, the creativity matrices (section III) would help explaining why some parameter modifications made sense.

Some works try to enhance individual creativity with assisting tools [27]. This is a P-creativity approach [3], where the tools present sources of inspiration and analogies to human operators, something in line with the ideas we presented in section II. That work did not propose a specific framework, unlike our work that bases upon a previous architecture with the goal of extending it with new methods. For instance, variations of brainstorming (section VI) could be applied to either generate analogies or new sources of inspiration (e.g. lateral thinking, Section IX).

Sosa and Gero [28] propose a socio-cognitive framework to analyse the interaction between designers and social groups to alter an artefact. The building block is a socio-cognitive agent, which has a first class representation of social norms. Agents playing the role of adopters (consumers of opinions) and promoters (opinion leaders) are embodying the idea of creativity in this construction. Their interaction propagates ideas on how to modify artefacts. Compared to their work, which performs simulations, we intend to produce real creative processes following the formulas expressed in the paper. These formulas applied to [28] would tell that further work could include vertical thinking techniques (section IV) to design ways to address modifications in the artefacts. Also, additional variations

could be generated using creativity matrices, since they permit to draw analogies and derive variations (section III).

MASTER (Multi-Agent System for Text Emotion Representation) is another experiment where each agent has a digital emotional state and can influence others by reciting poems [29]. The work focuses on the poetry generation domain and incorporates techniques to generate poems, but also to interpret them. The reaction of the listener gives clues to the speaker to alter the poem and get a higher effect. Our approach is more generic, although our first experimentation focuses on the drawing domain. Unlike [29], we have not defined specific semantics, but this allows for generalizing the results to other problems. The scheme applied in [29] could be enriched with more tactics, such as generating lots of poems to have more feedback (section V.C) or the multiplying effect if a poet agent reuses successful poems for a number of agents (section V.D).

Mendez et al [30] use agents to create stories. There are director agents, to direct the story plot, that create new character and object agents, set the motivation for characters, and take care that characters do not perform undesired actions. Character agents use affinity models to regulate interactions between them. Our framework captures this process as a brainstorming, maybe a stop-and-go brainstorming (section VI.A) because the director agents puts barriers to how the story progresses and creates shifts in the plot.

XIII. Conclusions

In this work, the most important techniques that stimulate human creativity have been identified and analysed. Based on this analysis, the corresponding and appropriate parallels have been suggested for the conceptual design of an agent-based creative system. This system considers two groups of agents, one that uses some conventional problem solving techniques, and a second one that generates new paths by using creative methods, which are inspired by those used by humans. These new paths can be then executed and evaluated by the problem solving group in order to determine whether they drive to effective and potentially creative solutions that can be shown to the user.

A first prototype of a MAS using these ideas was reported in [1], by using the INGENIAS agent methodology [31], although it does not implement all the features defined in this paper. That MAS is being developed for specific case studies (e.g. the design of a chair), and assisting the user in their creative process. It starts with the identification of the knowledge representation model of the domain, with attributes and relationships as shown in the example above (see Fig. 7). This can be supported by the use of ontologies. As well, the agents in the knowledge processing system requires, besides the internal behaviour of each type of agent, the organization and information of them.

An issue that has not been developed yet is the way of determining the degree of creativity of the solutions generated by the system. This is questionable because it is already difficult (if not impossible) to determine it for human creations (this has been already discussed in several works, as it is mentioned in the introduction). It would be possible to define some metrics for it [32], and these can be checked by a new group of agents, in a similar way to some approaches that have been discussed in section XII. However, it is possible to determine whether the generated solutions satisfy requirements that define the problem, and this is the purpose of the problem solving group in our framework. At the end, the human being watching the results will determine which ones like more or consider more creative.

References

[1] L. De Garrido, J. Gomez Sanz, J. Pavón, "Agent-based Modeling of Collaborative Creative Processes with INGENIAS," *AI Communications*, vol. 32, no. 3, 2019, pp. 223-233.

[2] M. Boden, "Creativity and artificial intelligence", in *Artificial Intelligence*, vol. 103, no. 1-2, 1998, pp. 347-356.

[3] M. Boden, "Creativity," in *Artificial Intelligence,* Academic Press, 1996, pp. 267-291.

[4] K. Jennings, "Developing creativity: Artificial barriers in artificial intelligence," in *Minds and Machines*, vol. 20, no. 4, 2010, pp. 489-501.

[5] R. A. Brooks, "Intelligence without reason". Massachusetts Institute of Technology. *Artificial Intelligence Memo No. 1293*, 2010.

[6] M. Rhodes, "An analysis of creativity," in *Phi Delta Kappan*, vol. 42, no. 7, 1961, 305–310.

[7] A. Jordanous, "Four PPPPerspectives on computational creativity in theory and in practice," in *Connection Science*, vol. 28, no. 2, 2016, pp. 194-216.

[8] C. Lamb, D.G. Brown, and C. Clarke, "Evaluating computational creativity: An interdisciplinary tutorial," in *ACM Computing Surveys* vol. 51, no. 2, 2018, pp. 1-34.

[9] G. A. Wiggins, "A preliminary framework for description, analysis and comparison of creative systems," in *Knowledge-Based Systems*, vol. 19, no. 7, 2006, pp. 449-458.

[10] M. Martin, and K.I. Voigt, "What Do We Really Know about Creativity Techniques? A Review of the Empirical Literature," *The Role of Creativity in the Management of Innovation*, 2017, pp. 181-203.

[11] S. A. Leybourne, "The Creativity Matrix: Balancing Architectural and Process Creativity in Project-based Management," in *IRNOP Conference*, Oslo, Norway, 2013, pp. 17-19.

[12] V. Tang, J. Luo, "Idea matrix and creativity operators," in *DS 75-7: Proceedings of the 19th International Conference on Engineering Design (ICED13), Design for Harmonies, Vol.7: Human Behaviour in Design*, Seoul, Korea, 2013, pp. 19-22.

[13] M. A. Runco, *Problem Finding, Problem Solving, and Creativity*, California, USA, Greenwood Publishing Group, 1994.

[14] I. Wopereis, E. Derix "Seeking Creativity: A Case Study on Information Problem Solving in Professional Music," in *Kurbanoğlu S. et al. (eds) Information Literacy: Key to an Inclusive Society, ECIL 2016, Communications in Computer and Information Science*, vol. 676, Springer, 2016.

[15] J. G. Rawlinson, *Creative thinking and brainstorming*, Routledge, 2017.

[16] A. F. Osborn, "Creative Thinking," in *American Association of Industrial Nurses Journal*, vol. 6, no. 9, 1954, pp. 23-25.

[17] M. E. Raichle, "The brain's default mode network," in *Annual Review of Neuroscience*, vol. 38, 2015, pp. 433-447.

[18] L. De Garrido, *Applications of Artificial Intelligence in the composition of architectural objects*. PhD Thesis, Escuela Técnica Superior de Arquitectura, Universidad Politécnica de Valencia, Spain, 1989.

[19] E. De Bono, *The Use of Lateral Thinking*, London, Jonathan Cape Ltd. 1967.

[20] E. De Bono, *Six thinking hats*, London, UK, Penguin, 2017.

[21] F. Garijo, and L. De Garrido, "A knowledge based system for house design," in *Proceedings of the 1988 IEEE International Conference on Systems, Man, and Cybernetics*, 1988, pp. 806-809.

[22] O. López-Ortega, "Computer-assisted creativity: Emulation of cognitive processes on a multi-agent system," in *Expert Systems with Applications*, vol. 40, no.9, 2013, pp. 3459-3470.

[23] C. Olszak, and T. Bartus, "Multi-agent approach in designing of organizational creativity support," in *The European Conference on Information Systems Management*, Academic Conferences International Limited, 2015, pp. 93.

[24] L. Macedo, and A. Cardoso, "Using Surprise to Create Products that get the Attention of other Agents," in AAAI Fall Symposium, 2011, pp. 79-84.

[25] M. d'Inverno, and M. Luck, "Creativity through autonomy and interaction," in *Cognitive Computation*, 4(3), 2012, pp. 332-346.

[26] G. R. Greenfield, "Computational aesthetics as a tool for creativity," in *Proceedings of the 5th conference on Creativity & cognition*, 2015, pp. 232-235.

[27] U. Lösch, J. Dugdale, Y. Demazeau, "Requirements for supporting individual human creativity in the design domain," in *International Conference on Entertainment Computing*, Springer, 2009, pp. 210-215.

[28] R. Sosa, J. S. Gero, "A computational framework for the study of creativity and innovation in design: Effects of social ties," in *Design Computing and Cognition'04*, Springer, 2004 pp. 499-517.

[29] A. Kirke, E. Miranda, "Emotional and multi-agent systems in computer-aided writing and poetry," in *Proceedings of the Artificial Intelligence and Poetry Symposium*, 2013, pp. 17-22.

[30] G. Méndez, P. Gervás, and C. León, "A model of character affinity for agent-based story generation," in *9th International Conference on Knowledge, Information and Creativity Support Systems*, Limassol, Cyprus, vol. 11, 2014.

[31] J. Pavón, J. J. Gómez-Sanz, R. Fuentes, "The INGENIAS methodology and tools," in *Agent-oriented methods*, IGI Global, Idea Group Publishing, 2005, pp. 236-276.

[32] J.J. Shah, S.M. Smith, N. Vargas-Hernandez, "Metrics for measuring ideation effectiveness," in Design Studies, vol. 24, no. 2, 2003, pp. 111-134.

Luis De Garrido

Architecture Department. Universitat de València. Spain. National Association of Sustainable Creative Architecture Research. Spain. PhD degree in Architecture (Universidad Politécnica de Valencia, 1989) and Doctor Honoris Causa (Universidad San Martin de Porres, 2019). Currently he develops a professional activity as an architect, professor and independent researcher.

Jorge J. Gómez Sanz

Institute of Knowledge Technology. Universidad Complutense de Madrid. Spain. Associate Professor in the Universidad Complutense de Madrid. He holds a degree in software engineering and a PhD (2002). His PhD introduced a new methodology for the development of Multi-Agent systems: INGENIAS. He has focused his research in agent oriented software engineering.

Juan Pavón

Institute of Knowledge Technology. Universidad Complutense de Madrid. Spain. PhD degree in Computer Science (Universidad Politécnica Madrid, 1988). Currently he is Full Professor at Universidad Complutense Madrid, where he leads the GRASIA research group and the Institute of Knowledge Technology. His main areas of interest focus on the application of Artificial Intelligence in multidisciplinary projects with social value, such as assistive technologies, health monitoring, ambient assisted living, smart cities, education, computational creativity, and tools to support Responsible Research and Innovation (RRI).