Towards a Computational Case-Based Model for Creative Planning

Luís Macedo (*), Francisco C. Pereira (**), Carlos Grilo (**), Amílcar Cardoso (**)

(*) Instituto Superior de Engenharia de Coimbra, 3030 Coimbra, Portugal

(macedo@alma.uc.pt)

(**) Dep. Eng. Informática, Univ. Coimbra, Polo II, 3030 Coimbra, Portugal

(francisco@alma.uc.pt, grilo@alma.uc.pt, amilcar@dei.uc.pt)

Abstract: This paper describes a computational case-based model for the creative planning process. Our approach is inspired in Wallas' model for the creative process in that we consider that creativity involves a sequence of four stages: preparation, incubation, illumination and verification. Preparation includes problem acquisition and assimilation of background knowledge, which is represented by cases, i.e., documented past experiences. With the aim of achieving a flexible knowledge representation, as a means to potentiate specific creative abilities like Fluency, Synthesis and Analysis, we structure each case as a network of hierarchically and temporally related case pieces. These case pieces can be considered individually, providing better recombinations of them. These recombinations, rather than made by chance, are guided by those hierarchical and temporal case piece relations (or explanations). We explain the role of opportunistic knowledge acquisition at the incubation stage. We sustain that illumination may comprise recursive calls of the sequence of the first three stages.

This computational model is implemented in the system INSPIRER (ImagiNation[1] taking as Source Past and Imperfectly RElated Reasonings). An application in musical composition domain is presented. We also show how a musical composition task may be cognitively modelled and treated as a planning task. We also present a short example illustrating how INSPIRER generates music.

1. Introduction

Creativity is an issue that has been challenging many cognitive science researchers. Although some progress has been achieved in this field, there are many opposite and no complete theories about it.

In the literature, two main concepts are referred as related to the act of creation: to bring into existence something original and novel which did not exist before; to give an original way of existence to something that already exists. However, everybody seems to agree that one cannot originate new things out of nothing. Besides originality, valuability is also referred by the authors as an important characteristic of a creative product (Ibáñez, 1991). Cabezas (1993), for instance, states that creativity results from the combination of previous ideas, experiences, elements, phenomena, images, realities, etc., in a new, original and useful way.

Although there are several explanation models for the creativity process, we are interested in those entailed to problem-solving (also called cognitive models). From the main percussor problem-solving models (Brown, 1989), i.e., Good Problem Solving (J. Dewey, 1910), Creative Production (Wallas, 1926), and Invention (Rossman, 1931). Wallas' model has much acceptance in cognitive research community. He proposed that creativity involves four sequential steps: preparation, incubation, illumination and verification.

Spiro defended that flexible knowledge is a prerequisite for knowledge restructuring and hence for creativity (Spiro, Vispoel, Schmitz, Samarapungavan & Boerger, 1987). He sustains that flexible knowledge representation is one in which fragments of knowledge are represented in a way that allows to be reassembled into new knowledge structures. Knowledge-based retrieval systems (Koton, 1989) are a consequence of combining nearest neighbour and knowledge-guided techniques. These systems are characterised by the use of domain knowledge for the construction of explanations for why a problem had a particular solution in the past.

---

[1] We consider that imagination is the mental creation of ideas while creativity involves the imagination of ideas and its materialisation.

Case-Based Reasoning (CBR) systems represent this domain knowledge by cases (Kolodner, 1993). They are appropriate for domains where a strong theory does not exist but past experience is accessible.

A plan is a specific sequence of steps (or actions) with the aim of a goal achievement. Case-Based Planning (CBP) systems (Hammond, 1989; Veloso, 1992) reuse past sequences of actions from past plans to construct new ones. There are some case-based models for the creative process. Linda Wills and Janet Kolodner (Wills & Kolodner, 1994) consider three steps in creative design: enumeration of several alternative solutions; re-description and elaboration of problem specifications; and evaluation of proposed solutions. Paulo Gomes et. al.(Gomes, 1996) proposed a case-based model for creative processes in which creativity is a result of searching on spaces of cases increasingly further away from the target problem.

In this paper we will focus on a computational case-based model for the creative planning process. Our approach to creativity is based on Wallas' model, although we adopt a non-linear, recursive, execution of the four sequential stages. We think that the potential solution for the problem obtained at the illumination stage may be composed by another problems and each one of them may conduct to a sequence of three stages: preparation, incubation and illumination. The solution proposed in each one of these illumination stages may also include the resolution of another problems, the sequence of the first three stages being repeated for each one. The global validation and revision of the entire solution is made at the verification stage.

The previous knowledge necessary to the creative process is represented by cases. Each case is a structured network of implicitly and explicitly, hierarchically and temporally related case pieces. We think that considering cases as sets of related pieces provides a more flexible knowledge representation, and hence, facilitates the recombination and reordering of the original structures into newer ones.

Our approach to creativity is presented in the next section. In section 3, we introduce our computational model for creative planning, subdivided in four phases. Section 4 presents an application in the music composition domain. We also explain how the music composition process may be seen as a planning task and how it is executed by musicians in reality. A short example illustrating how music is generated by INSPIRER is presented in section 5. At last, a conclusion about our work is made in section 6.

2. Our approach for creativity

We think that creativity is characterised by a connection, combination, association or integration of different (and possibly contrary) things in the same product (the creative product) directly or using metaphoric and analogic techniques. However, to be considered creative, the product must be (Cabezas, 1993):

I. novel, i.e., not common, not conventional and not frequent. E.g., the Copernicus's theory was a new scientific product, which no one proposed before;

II. unpredictable, i.e., creativity activity uses and changes in an unpredictable way what already exists. This means creative products are not totally and not perfectly causally explained. They are a consequence of inspiration or irrational activity. E.g., it is not possible to explain why J. S. Bach created *El magnificat* in that way and not in other;

III. singular, i.e., it must be unique, idiosyncratic, peculiar and personal. E.g., every musician or writer has a particular style, which is preserved in all his/her products;

IV. surprising, i.e., it must cause a psychological and unexpected astonishment. E.g., a creative poem induces in the reader (and also in the author) a sentimental and emotional elevated shock;

V. useful, i.e., it must be something that provides a better existence to someone (solving a problem), or making he/she feel emotionally better (being aesthetic).

We based our explanation model for the creative process in the Wallas' one. In our opinion, its well adequate explanation of the creative process, judging by the reports of creative persons about their creative experiences, has weighted on our choose. We think it may be adapted specifically to creative planning, as it is a general approach. Thus we consider that a creative product (plan, design, etc.) is obtained performing the following steps:

1) Preparation. This phase includes: (i) a formulation of the problem in sense of knowing what is to be solved; (ii) an accumulation or assimilation of knowledge, to which we call background knowledge, necessary to create something.

2) Incubation. This phase corresponds to the generation and formulation of possible solutions. This process can be unconscious or partially conscious. During this phase, the problem is being unconsciously pursued and the flexibly organised background knowledge, acquired during the anterior phase, is being restructured into new schemata, i.e., new mental structures are created by recombinations and reorderings of the original knowledge (Armbruster, 1989). In our approach, this process of recombination of knowledge fragments is performed in the following way. Considering that π is the currently being constructed solution (at the beginning is empty), and π_i a place on solution π in which is lacking a

knowledge fragment: (i) selection and addition to π of a fragment that is associated in the background knowledge with similar problems. This selection is made taking into account the similarity between the context of π_i on solution π, and the context of the considered fragment on the original knowledge. Thus, the combination has more probabilities to be more creative if the selected fragment is the one with less context similarities. However, this way the probabilities to be a bizarre combination also grow; (ii) validation of the addition of the new fragment to the solution.

Simultaneously to this knowledge restructuring, new knowledge may be acquired which may lead to new recombinations or may fill some lacking piece of knowledge, and then, the solution may be completed conducing to the illumination stage, or at least augmented, just like putting a lacking piece on an incomplete puzzle, and the incubation proceeds. This kind of knowledge is called opportunistic knowledge and the process, opportunistic reasoning (Hammond, 1988). This is particularly clear in situations where illumination or insight comes when mental work is not pursuing the problem, but suddenly, receives news experiences that directly or by analogical reasoning constitute or conduce to the problem solution. E.g., Newton discovered the Universal Gravity Laws seeing an apple falling down from a tree.

3) Illumination. In this stage the solution is consciously proposed. This solution may imply the decomposition of the problem in sub-problems, which will be recursively pursued through the first phases of the creative process.

4) Verification. In this stage the creative properties (novelty, usefulness, etc.) of the solution are tested and some revisions and adaptations are made when necessary. If the solution is still considered as a non-creative product then it may be refused and all the sequence may be repeated from the beginning, trying to find a new knowledge recombination.

3. A computational model for creative planning

In this section we describe our creative planning process from a computational point of view.

3.1. Preparation

3.1.1. Background knowledge

As was said above the creative process is based on previous knowledge which we represent by cases. Since we deal with creative planning, cases are plans.

Within our approach a case plan is a set of goals and actions organised in a hierarchical way (Figure 1): a main goal (the main problem) is refined into sub-goals (the sub-problems), and so on, until reaching the actions that satisfy the goals.

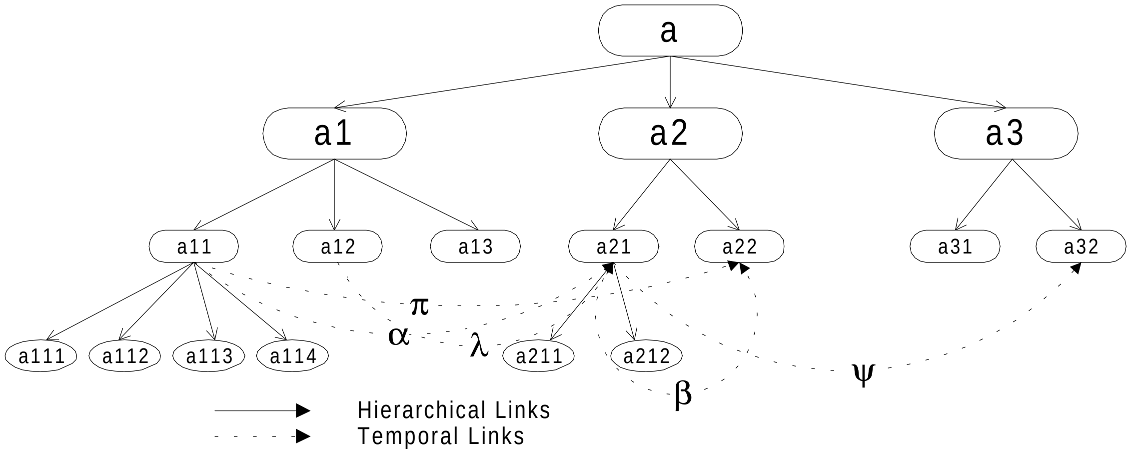

Figure 1 - Case structure.

In our model, each node of the hierarchical structure corresponds to a case piece. To complete the case structure, there are links between case pieces, representing causal justifications, or explanations. Some of these links maintain the hierarchical case structure, others reflect temporal relations between case pieces. Thus the existence of a case piece in a plan case is causally explained by several case pieces of the same plan case.

A measure of importance (strong, weak or medium) is given to each explicit link, according to its weight in the explanation of the consequent case piece.

Considering the hierarchical links only (represented in Figure 1 by continuous arrows), the inherent meaning of the represented structure is: *a*, the main goal of the plan (or the main problem) is achieved by sequentially achieving sub-goals (sub-problems) *a1*, *a2* and *a3*. Each one of these sub-goals is also broken up into other sub-goals. For example, *a1* is broken up into *a11*, *a12* and *a13*, and *a2* into *a21* and *a22*. To achieve the goal *a11* the actions *a111*, *a112*, *a113* and *a114* must be sequentially executed by this temporal order.

Besides being explained by the goal-refinement process, through hierarchical links, a case piece may also be explained through a temporal link (represented in Figure 1 by discontinuous arrows). For example, *a21* (sub-goal of *a2*) is a consequence of case pieces *a11* and *a12*, which is represented by the temporal links labelled α and λ, respectively.

Since the case pieces form a tree-like structure we adopt the tree characteristic terminology to facilitate the description of our approach. Thus, we say that a case piece is a node and belongs to a level (e.g., in Figure 1, case pieces *a1*, *a2* and *a3* (sub-goals of goal *a*) belong to level 1; case piece *a* (the main goal) belongs to the level 0). We also consider that father, son, brother, etc., relations exist between case pieces (e.g., in Figure 1, case piece *a* is father of case piece *a1*, *a1* is son of *a* and brother of *a2* and *a3*).

A case piece has seven types of information describing its relevant aspects: a name that uniquely identifies the case piece; the name of the case to which the case piece belongs; the case piece address; the constraints, i.e., the pre-conditions which must be satisfied for the case piece to be accepted for retrieval; a set of attribute/value pairs describing it; the antecedents and the consequents.

The address of a case piece represents its hierarchical (level) and temporal position on the case (for more details see Grilo, Pereira, Macedo & Cardoso, 1996). It also embeds in its syntax pointers to its ascendants and brothers.

Antecedents and consequents are links that follow, respectively, from and to other case pieces. Antecedent links explain the existence of the considered case piece (e.g. in Figure 1, *a21* is explained by *a11* and *a12*, by the links labelled α and λ, respectively, and by *a2* through a father link). Consequent links show how a case piece explains the existence of other case pieces (e.g. in Figure 1, *a21* partially explains *a22* and *a32* by links β and ψ, respectively, and *a211* and *a212* by father links).

Each antecedent or consequent link is classified into another two main kinds of links, hierarchical and temporal.

Hierarchical links reflect the case pieces refinement (e.g., in Figure 1, there is a hierarchical link between goal *a* and goal *a1* because *a1* is a subdivision of *a*).

A temporal link expresses a causal explanation between two temporally disjunct case pieces (e.g., in Figure 1, case piece *a21* is explained by case piece *a11*). The explanation embeds the temporal relation between the case pieces.

Sometimes the type of relation between antecedent fact(s) and the consequent one may be unknown. This lack of a complete theory is common in several domains like psychological, medical, artistic, etc. (Bento, Macedo & Costa, 1994). This idea leads to another classification of the links between case pieces: we say that a link between the case pieces *a* and *b* is explicit if the concrete relation between *a* and *b* is known, and implicit if not. In Figure 1, *a13* implicitly (and temporally) explains *a21*. There is not a concrete link between them, but it is coherent to assume that the existence of *a21* is partially due to the previous occurrence of *a13*. We may also say that *a* implicitly explains *a21*, although there is not a direct relation between them.

We call the case piece *context* to the set of case pieces that surrounds it. We distinguish eight types of contexts according to the kind of link existing between the case piece considered and the surrounding ones. Thus, each one of these surrounding case pieces is included in one of the following contexts (the name of the context reflects the classification of the link to the case piece): antecedent-hierarchical-implicit context, antecedent-hierarchical-explicit context, antecedent-temporal-implicit context, antecedent-temporal-explicit context, consequent-hierarchical-implicit context, consequent-hierarchical-explicit context, consequent-temporal-implicit context or consequent-temporal-explicit context.

For example, in Figure 1, the contexts of *a21* are: antecedent-hierarchical-implicit context = {*a*}; antecedent-hierarchical-explicit context = {*a2*}; antecedent-temporal-implicit context = {*a13*}; antecedent-temporal-explicit context = {*a11*, *a12*}; consequent-hierarchical-implicit context = {}; consequent-hierarchical-explicit context = {*a211*, *a212*}; consequent-temporal-implicit context = {*a31*}; consequent-temporal-explicit context = {*a22*, *a32*}.

Since there is not any direct link between implicitly related case pieces, it is necessary to define a frontier to limit the number of case pieces of the implicit contexts. We assume that this frontier involves the nearest case pieces. To each implicit type of context, we defined a user-configurable parameter with the maximum distance a case piece may be to belong to a context of that type.

3.1.2. Problem

A new problem to be solved by the system may comprise a set of linked case pieces. At least the main goal (the root case piece) must be included, with its name, address, constraints and attributes information instanciated.

The meaning associated to a problem description composed by the main goal is the following: the system must find a structured plan solution to achieve the goal; the solution must satisfy the goal's constraints. If the problem also includes sub-goals or actions with the same instanciated information types, then the meaning of the problem description is augmented by the following: the system must find a structured plan solution to achieve the goal; the solution must satisfy the goal's constraints and must achieve the specified sub-goals and perform the specified actions. Thus a problem may be a partial structured solution given by the user. The system just have to coherently complete it.

3.2. Incubation

At this stage the main goal is considered to be solved by recombining sub-goals or actions of previous cases. Since the complete information about this goal is not already known (for example, the number of sub-goals that are necessary to achieve it or the links that follow from it may be unknown at this point), the main goal that best matches the considered one is retrieved from a case in memory. The next step is retrieving the main goal's sons from memory, starting by the oldest, assuming its context (currently, the retrieved main goal) and its address as indexes.

The retrieving of a case piece from memory involves the following steps (given the context and the address of the next free position on the new case solution):

1) selection of the candidate case pieces from memory, eliminating those which are incompatible with the constraints of the free position's father, and those which do not belong to the same level of the free position;
2) application of a similarity metric[2] taking into account the given context and address, and ranking of the case pieces selected in step 1;
3) selection of a case piece according to the criterion established by the user to the current hierarchical level (each level has selection criterion. For example, in level 1 the selected case piece must be the one with more similarity metric value, but in level 4, it must be the one with the less similarity metric value);
4) validation of the addition to the solution of the selected case piece.

From the above algorithm it can be seen that the weighted similarity metric used for selection of a case piece takes into account the similarity between the context and the address considered in the new case under construction and the context and the address of the candidate case piece.

As was said above, we consider eight types of contexts. Obviously the similarity metric must give different weights to different context similarities. For example, it must give a bigger weight to explicit link's similarities than to implicit ones. The similarity metric also favours the case pieces whose addresses are more similar to the free position's address.

The similarity metric value for a case piece is expressed in percentage. Thus, when a case piece has a similarity value equal to x%, means that its context and address is x% similar to the context and address of the free position. Selection of a case piece involves the computation of a similarity metric value for it and the consideration of the selection criterion. This means that, first, the candidate case pieces are ranked by their similarity metric value, and then, the established criterion for that level determines which case piece is selected. This criterion reflects the degree of originality needed in that level. For example, the user may want that the level 1 of the solution must have null originality, because at this level new solutions are always similar to old ones. However, he/she also may wants that solutions, in level 5, must be, at maximum, 50% original. Therefore, in this level, the candidate case pieces with a similarity value lesser than 50% are not considered to be selected. The first one tried to be selected is the one with a similarity value equal or higher than 50%. This means that the case pieces recombination is not made by chance, but instead considering the needed originality.

After a case piece is selected, it is submitted to a validation process consisting in the verification of incompatibilities between the selected case piece and the partially constructed solution for the given problem. At this point, there may be links that follow from earlier case pieces, pointing to the free position. We call them *suggestions*, as they correspond to proposed but not definitive links. If an incompatibility exists between a suggestion and an antecedent link of the selected case piece there are three choices: (i) try to adapt it, relaxing the validation by ignoring the less important of the incompatible links (e.g., if the suggestion is strong and the antecedent link of the selected case piece is weak, the validation step substitutes the second link by the former one in the selected case piece, and then this case piece is added to the new case); (ii) if it was not possible to adapt it, select another one (taking into account the criterion for that level) and apply the validation step to it; (iii) if there are not alternative case pieces for the free position, then the problem being solved is suspended or putted in standby. When the system acquires new knowledge the standby problem(s)

[2] E.g., Bento's quantitative metric (Bento & Costa, 1994).

is(are) activated and tried to be solved, with the expectation that this opportunistic knowledge may be the key for the solution.

3.3. Illumination

At this stage, the solution constructed in the incubation stage is proposed. However, as we said the solution may include problems. Each one of these problems initiates another sequence of three stages (preparation, incubation, illumination) until its solution is founded. This solution may also be composed by another problems, and so on.

3.4. Verification

At this stage, the entire solution is submitted to a validation and test about its originality and usefulness. If necessary some revisions are made. At this point, this stage is performed by the user.

4. An application in Musical Composition Domain

As studied by Lerdahl and Jackendoff (1983), Balaban (1992) and Honning (1993), music is a domain in which "structure", "hierarchy" and "time" are more than occasional keywords. Music is indeed a highly structured and organised world. As stated by Balaban, any music can be represented by a hierarchy of temporal objects (an object associated with a temporal duration), in such a way that each one has, as descendants, a sequence of sub-objects that starts and ends at the same start and ending point as the object's. Figure 2 shows an example.

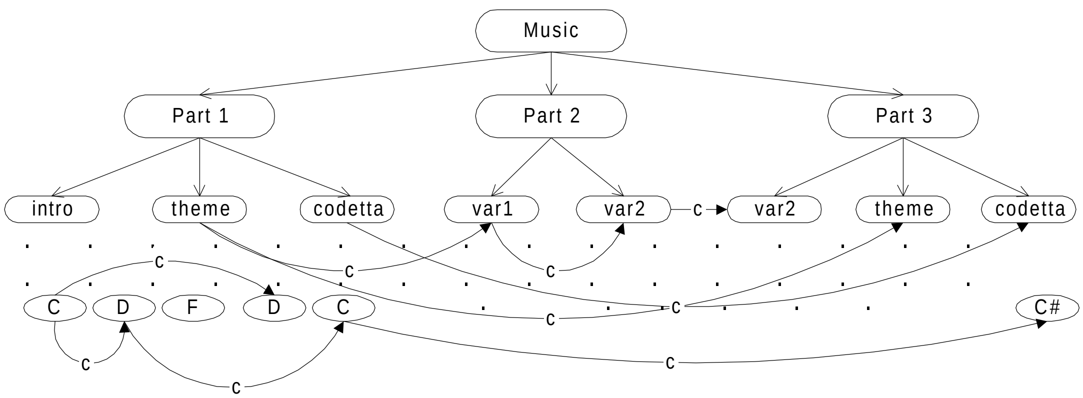

Figure 2 - A case in the Music domain.

There are important relations in music, since many musical objects may be causally explained by a transformation (like "repetition", "variation", "inversion", "transposition", etc.) of some other object. These causal relations are represented through temporal links ('c' links in Figure 2) which correspond to the associated transformations (e.g. the temporal link between "theme" of "part1" and "var1" of "part2" may represent a "variation" transformation which, when applied to "theme" originates "var1").

The goal of our application is to use music analysis as foundation for a generative process of composition, providing a structured and constrained way of composing creative pieces, although keeping the essential traits of the composer's style. We use analysis of music pieces from a seventeenth century composer.

We have concluded that considering music as a plan, with the organisational characteristics described earlier, and the act of composing as CBP, might be an interesting way of generating new music from old ones. In fact, music structure has the basic conditions to be considered as a normal plan structure.

The use of four stages to perform a musical creation is corroborated by musicians' experience. In fact, they start by considering the main problem (e.g., a sonata), which is solved with a sequence of another set of problems (e.g., Part1, Part2 and Part3). This problems are also solved by an analogous way, and so on, until is reached the music notes. After this, the entire music is validated and , if necessary, some revisions are made.

5. An example

In this section we briefly illustrate the musical generation process used by INSPIRER.

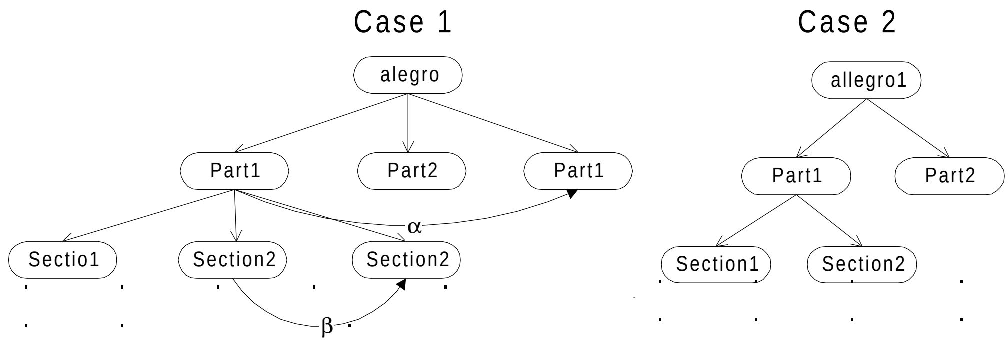

Figure 3 - Cases in memory.

Preparation. The system was seeded with two musical cases represented in Figure 3[3]. The problem given to the system (represented by the PROLOG fact case_node(new_case, allegro, 0, [], [ton='I',comp=2/4],_,_)) is to come up with a music sonata characterised by having binary measure and tonality 'I''.

Incubation. First, a case piece with more similarities with the one represented in the problem is retrieved from a case in memory.

If there were not any case in memory then the system put the problem in standby, waiting for opportunistic knowledge.

For the current problem, and considering that INSPIRER has case 1 and case 2 in memory, the system retrieved the main goal of case 1 since it is the one with more similarities with the goal proposed as problem. At this point, the solution is the one presented in Figure 4 - (i).

Next step is finding a sequence of three nodes which jointly will form a solution for the problem. The first new case free position tried to be filled is the one with address 0:0. The system found in memory the following set of candidate pieces, which are ranked by similarity metric value with the new case's free position of address 0:0: Part1 from case 1, Part 1 from case 2, Part2 from case 1, Part 2 from case 2. The selection of one of these pieces depends on the used criterion. In this level the criterion is to select the most similar. Therefore Part1 from case 1 is selected. However, if the criterion was to select the less similar, then Part1 from case 2 was selected. But, before its addition to the solution, this piece is submitted to a compatibility test. Since it has not any incompatibilities with other pieces of the current solution, it is added to it (Figure 4 - (ii)).

Next step is retrieving a node for new case free position with address 1:0, and then to new free position with address 2:0. The system chose Part2 from case1 and Part1 also from case 1, respectively (Figure 4 - (iii)). Thus the solution is reached.

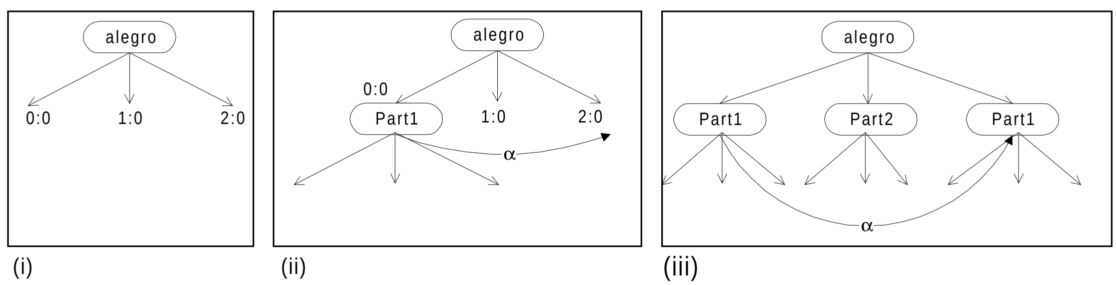

Figure 4 - New case generation.

Illumination. The solution is the temporal sequence of Part1, Part2 and Part1 (repetition).

However, these pieces are also unsolved problems. Consequently, each one is included in a new preparation stage. The system starts by Part1 to which found the following solution: Section 1 from case1, Section2 from case 1, and Section2 from case 1 (repetition). Then it takes Part2 as a problem and by a similar way achieve to it a solution.

These solutions also include problems and the process is recursively repeated until reaching the actions (in this application, the actions are the music notes).

Verification. The entire solution presented to the user by the system is validated by an expert. Some notes which are incorrectly or anaesthetically combined are adequate to avoid it.

---

[3] Because of the extent of musical cases we have to represent incomplete ones.

6. Conclusions

We have presented an approach to a computational model for creative planning, taking as source cases of past plans. With the aim of dealing with flexible knowledge, cases are split into pieces, providing a wide variety of recombinations of these pieces. These recombinations, rather than made by chance, are guided by the similarities between the context (defined by the links) and the address of the case pieces in memory and of the free position on the under construction solution.

As stated by Wallas, the creative process involves four stages. However, we defend that opportunistic knowledge takes an important role in the incubation stage, and that the potential solution proposed in the illumination stage may comprises a problem decomposition.

As shown, musical composition task can be considered as a planning task and is an appropriate domain to our creative approach, judging by the reports of musicians about their creative experiences. However, other domains like story or screenplay generation are also possible applications for INSPIRER.

Acknowledgements

We would like to thank to Anabela Simões and António Andrade, teachers at the Coimbra School of Music for their valuable contribution, and the anonymous reviewers of this paper for their helpful comments.

References

Armbruster, B., (1989). Metacognition in Creativity, in John Gloves, Royce Ronning & Cecil Reynolds (Eds.), Handbook of Creativity, New York and London: Plenum Press.

Balaban, M., (1992) - Musical Structures: Interleaving the Temporal and Hierarchical Aspects in Music. In Understanding Music with IA: Perspectives in Music Cognition, MIT Press, pp. 110 - 138.

Bento, C. and Costa, E., (1994). A Similarity Metric for Retrieval of Cases Imperfectly Explained. In Wess S.; Althoff, K.-D.; and Richter, M. M. (eds.), Topics in Case-Based Reasoning - Selected Papers from the First European Workshop on Case-Based Reasoning, Kaiserslautern, Springer Verlag.

Bento, C., Macedo, L., & Costa, E., (1994). RECIDE - Reasoning with Cases Imperfectly Described and Explained, in Proceedings of Second European Workshop on Case-Based Reasoning, France.

Brown, R., (1989). Creativity: What are we to measure?, in John Gloves, Royce Ronning & Cecil Reynolds (Eds.), Handbook of Creativity, New York and London: Plenum Press.

Cabezas, J., (1993). La Creatividad. Teoría básica e implicaciones pedagógicas. Libreria Cervantes, Salamanca (in spanish).

Dewey, J., (1910). How we think. Referenced by Brown, R. T. (1989). Creativity: what are we to measure?. In Glover, J., Ronning, R. and Reynolds, C., Handbook of Creativity, Plenum Press, New York.

Gomes, P., Bento, C., Gago, P. & Costa, E., (1996). Towards a Case-Based Model for Creative Processes, in Proceedings of ECAI-96.

Grilo, C., Pereira, F., Macedo, L. & Cardoso, A., (1996). A Structured Framework for representing Time in a Generative Composing System. (unpublished)

Hammond, J., (1988). Opportunistic Memory: Storing and recalling suspended goals, in Proceedings of a Case-Based Reasoning Workshop, Florida.

Hammond, K., (1989). Case-Based Planning: Viewing planning as a memory task, Academic Press, Inc., New York, New York, 1989.

Honning (1993). Issues in the representations of time and structure in music. Contemporary Music Review, 9, pp. 221-239.

Ibáñez, R. M., (1991). Definicion de la creatividad. In: *Manual de la Creatividad*, Marin, R. and de la Torre, S., Vicens Vive, Barcelona (in spanish).

Kolodner, J., (1993). Case-Based Reasoning, Morgan-Kaufmann, San Mateo, CA.

Koton, P., (1989). Using Experience in Learning and Problem Solving, Massachusets Institute of Technology, Laboratory of Computer Science (PhD diss., October 1988), MIT/LCS/TR-441.

Lerdahl, F. and Jackendoff, R. (1983). A Generative Theory of Tonal Music. Cambridge, Mass.: MIT Press.

Rossman, J., (1931). The psychology of the inventor: A study of the patentee, Referenced by Brown, R. T. (1989). Creativity: what are we to measure?. In Glover, J., Ronning, R. and Reynolds, C., Handbook of Creativity, Plenum Press, New York.

Spiro, R., Vispoel, W., Schmitz, J., Samarapungavan, A. & Boerger, A., (1987). Knowledge acquisition for application: Cognitiveflexibility and transfer in complex cognitive domains. Referenced in Armbruster, B., (1989). Metacognition in Creativity, in John Gloves, Royce Ronning & Cecil Reynolds (Eds.), Handbook of Creativity, New York and London: Plenum Press.

Veloso, M., (1992). Learning by Analogical Reasoning in General Problem Solving, PhD D thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA.

Wallas, G. (1926). The art of thought. Referenced by Brown, R. T. (1989). Creativity: what are we to measure?. In Glover, J., Ronning, R. and Reynolds, C., Handbook of Creativity, Plenum Press, New York.

Wills, L. & Kolodner, J., (1994). Towards More Creative Case-Based Design Systems, in Proceedings of AAAI-94.