A Hybrid Model of Reasoning by Analogy*

Boicho Nikolov Kokinov

1. INTRODUCTION

This chapter describes an attempt to model human analogical reasoning at the level of behavioral constraints (Palmer, 1989) (i.e., the aim is to develop a computational model which will reflect people's observable behavior).

In order to be more concrete, I will elaborate on reasoning by analogy only in a problem-solving task, although some of the proposals could still be valid in other kinds of tasks like explanation, argumentation, etc.

1.1. Dynamic Aspects of Human Reasoning

Most people can remember at least one occasion when they failed to solve a problem at their first attempt at it, but succeeded, and without great difficulties at that, if they had a second chance later. Moreover, people also happen to be unable to solve for a second time a problem they have successfully solved before. It is also quite common for people to find various solutions of one and the same problem in various occasions. As a rule, this variability and flexibility of human problem-solving behavior is ignored by models of analogy and problem solving in general.

* I am grateful to P. Barnev, E. Gerganov, S. Kerpedjiev, V Nikolov, and C. Castelfranchi for the valuable comments on a draft of the chapter as well as to all participants in the regular seminar of the Bulgarian Society for Cognitive Science for the relevant discussions. I am deeply indebted to the editors—Keith Holyoak and *John* Barnden—both for their useful advice and for their patience and helpfulness, as well as to all the five anonymous reviewers for their numerous thoughtful comments according to which this chapter has been improved. I thank V Nikolov and T. Kostadinova for their help in improving the text. I thank also my wife for her patience and continuous encouragement. This research was partially supported by the Bulgarian National Science Fund under Contract No 110/91 as well as by the Bulgarian Academy of Sciences (BAS) and the New Bulgarian University.

by several researchers (Hendler, i989a, 1989b, 1991; Lange & Dyer, 1989; Lange, MelzrWharton, & Holyoak, 1990; Dyer, 1991).

Usually there are objections to hybrid approaches as being too eclectic. Their critics claim that we have to explain human cognition by a single consistent approach; the question, of course, is whether this is possible at all. In my view it is clear that each real-world object or process is too complex to be fully described by a single formal theory or model, and therefore several different and possibly contradicting points of view are needed. This is especially true for such a complex object as the human mind (and human reasoning in particular). It might be the case that we need multilevel hybrid models in order to cover all aspects of human reasoning. An analogous conclusion about language is reached in Dyer (1991).

Multiview approaches are most often reduced to dualisms which, for this reason, are deeply rooted in human scientific thinking. Researchers often propose two opposite or orthogonal views on a single phenomenon to make its description more complete. The corpuscular and wave theories of light present a classical example of two different and complementary theories proposed in order to account for the inconsistencies in the properties of light under different conditions.

One of the basic dualisms in science is the discrete versus continuous points of view. The example above involves a dualism of that kind as well. It seems to me that, in order to account for the different properties of human reasoning, we have to incorporate the same fundamental dualism in the explanation. I consider symbolism and connectionism as particular realizations of the discrete and continuous paradigms, respectively, and I believe that we need to take both aspects into account.

Not every hybrid model, however, can be considered as a good realization of the dualistic principle. In my opinion, the right answer does not lie in developing models where separate modules correspond to separate phenomena or cognitive processes and are implemented within separate paradigms, like a connectionist model of perception and learning combined with a symbolic model of reasoning.; Instead, both aspects should be basic to the proposed cognitive architecture and contribute at every level to every cognitive process.

Sometimes unified theories emerge at a later stage of the development of such hybrid explanations. Referring to the example above, it was quantum electrodynamics that was developed as a single theory of light providing a unifying explanation of light's dualistic behavior, although at a rather abstract level. Following the same analogy, after constructing a hybrid model of human reasoning we could search for a more general theory explaining both aspects from a single point of view (but uncovered aspects will probably always remain, calling for explanations by different theories from other, complementary points °f view).

As one of very few exceptions, in the COPYCAT model of Hofstadter and his colleagues (Hofstadter, 1985; Mitchell & Hofstadter, 1990) the problem can be perceived in different ways in separate occasions, thereby generating different solutions. COPYCAT, however, provides a purely stochastic explanation and thus the factors contributing to the variability of problem solving are not clarified.

The explanation suggested in this chapter assumes that human reasoning (similarly to perception and language understanding) is actually context-dependent and thus evolves with the course of time. Here a broad notion of context is meant, including both the environment and the state of the reasoner's mind. In contrast, typical computational models of human reasoning consider the reasoner[1] in isolation from her environment and/or from her own thoughts and state of mind.

In this chapter an attempt is made to build a model which will somehow reflect the context and thus include this dynamic aspect of reasoning. For this reason memory is considered not as a static store but as a dynamic process running in parallel to all other reasoning components. This leads us to a hybrid (symbolic/connectionist) model with a high degree of parallelism.

In Section 2, the role of the preliminary setting (the internal context) in human problem solving and the way it develops over time is explored. In Section 3, the basic principles of the theory are stated. In Section 4, the cognitive architecture which underlies the model is outlined. In Section 5, the model of analogical reasoning is presented, and in Section 6, a simulation of human problem solving is described. Section 7 compares the present work to related research.

1.2. Hybrid Models: Eclectic or Consistent?

There are two main approaches to cognitive modeling in general: the symbolic approach and connectionism. The symbolic approach is still dominant in cognitive science and especially in modeling human reasoning as the latter requires elaborate structures, complicated syntactic manipulations and rich semantic representations, and for those the symbolic approach is well fit. On the other hand, there are aspects of reasoning which require dynamic modeling, high parallelism, competition, bringing together knowledge from various sources, etc., which are better mastered by connectionist models. None of the two approaches is ideal, however; in the recent years we have witnessed growing recognition of their limitations and the emergence of hybrid models developed

[1] Unless otherwise stated, the term *reasoner* is used in its general sense throughout this chapter (i.e., referring either to a human reasoner or to a simulation system). For convenience only, the reasoner is regarded as female. No specific restrictions on the reasoner's nature are implied, however.

1.3. Is Analogy Different from Deduction and Induction?

There is no general agreement between the researchers in the field about the nature of analogy. Michalski (1986, 1989) considered analogy as a two-step process with the first step being induction and the second one deduction. On the contrary, Holyoak and his collaborators (Gick & Holyoak, 1983; Holyoak & Thagard, 1989a) considered the induction step as a consequence of a successful analogy.

A widespread (and broadly accepted) definition of analogy is that it is a mapping between elements of a source domain and a target domain. Gentner (1989) stated that:

> analogy is a mapping of knowledge from one domain (the base) into another (the target), which conveys that a system of relations that hold among the base objects also holds among the target objects. Thus, an analogy is a way of focusing on relational commonalities independently of the objects in which those relations are embedded.... People prefer to map connected systems of relations governed by higher-order relations with inferential import, rather than isolated predicates, (p. 201)

Holland, Holyoak, Nisbett, and Thagard (1986) considered analogy as a second-order quasihomomorphism where the model of one real domain is considered as a model of another domain.

Clement (1988) restricted analogy only to the case where: (a) a subject, without provocation, refers to another situation B, where one or more features ordinarily assumed fixed in the original problem situation A, are different; (b) the subject indicates that certain structural or functional relationships may be equivalent in A and B; and (c) the related case B is described at approximately the same level of abstraction as A.

Eliot (1986) claimed that "research in many fields, including machine learning, cognitive psychology, and linguistics, does not make a clear distinction between the psychological phenomenon known as analogy and other types of problem-solving processes" (p. 17). The issue is whether such a distinction is either possible or necessary.

I do not believe that humans possess separate mechanisms for separate kinds of reasoning. I do believe that from a computational point of view, deduction, induction (generalization), and analogy are slightly different versions of a single uniform reasoning process. They differ in the outcome of the retrieval process and only with respect to this intermediate result and the correspondence between descriptions established during the mapping process, we can identify the reasoning process as deduction, induction, or analogy. In this way we can view the analogy case as the most general one with deduction and generalization at the two extremities—where the retrieved source and the target are related in a specific way, one of them happening to be a particular instance of the other.

Many researchers who model analogy separately suppose that, in the course of the reasoning process, an explicit decision to use analogy is made at the beginning, thus causing the application of the method of reasoning by analogy. For example, Wolstencroft (1989) stated explicitly that if we use one method in preference to any other one, we should have identified in advance that the chosen method will be the most likely to offer a solution, which is why he added an identification step to his model. In contrast with the above, I assume that typically the reasoning mechanism starts with its retrieval process and it is the result of the retrieval process which determines, at a later stage, the kind of reasoning used. Burstein and Collins (1988) and Collins and Michalski (1989), analyzing a set of protocols, also came to the conclusion that the kind of knowledge retrieved from memory drives the particular line of inference produced.

The present work is a part of a broader project aiming to elaborate and test the hypothesis about the uniformity of human reasoning. A uniform mechanism of human reasoning in a problem-solving task, called Associative Memory-Based Reasoning (AMBR), has been proposed (Kokinov, 1988b), and some experimental data supporting it has been obtained (Kokinov, 1990, 1992). As it is still in progress, in the current presentation I concentrate on the way AMBR models analogical reasoning, in spite of the fact that some of the considerations might be valid in other cases as well.

2. PSYCHOLOGICAL PHENOMENA TO BE MODELED

The general phenomenon to be modeled is that people do solve problems by analogy. This is, of course, well known from numerous experiments as well as from everyday life. We need, however, much more detailed information about they *way* people do it, which factors influence human performance and in what manner, and what kind of accompanying phenomena can be observed. There is a considerable shortage of psychological experiments that could provide answers to these questions, but there is some experimental data to be taken into account when modeling human analogical problem solving.

Analogical problem solving can be initiated by an explicit hint to use a particular case (provided by a teacher) as a source for analogy (Gick & Holyoak, 1980,1983), by a reasoner's explicit decision to try to solve a difficult problem by an (a priori unknown) analogy and generating (constructing) various sources by systematic transformations (Clement, 1988; Polya, 1954, 1957), or by spontaneous retrieval of a source from memory and noticing the analogy between this case and the target. In the present work only the last case is investigated: *the spontaneous use of analogy.*

1.3. Is Analogy Different from Deduction and Induction?

There is no general agreement between the researchers in the field about the nature of analogy. Michalski (1986, 1989) considered analogy as a two-step process with the first step being induction and the second one deduction. On the contrary, Holyoak and his collaborators (Gick & Holyoak, 1983; Holyoak & Thagard, 1989a) considered the induction step as a consequence of a successful analogy.

A widespread (and broadly accepted) definition of analogy is that it is a mapping between elements of a source domain and a target domain. Centner (1989) stated that:

> analogy is a mapping of knowledge from one domain (the base) into another (the target), which conveys that a system of relations that hold among the base objects also holds among the target objects. Thus, an analogy is a way of focusing on relational commonalities independently of the objects in which those relations are embedded____People prefer to map connected systems of relations governed by higher-order relations with inferential import, rather than isolated predicates, (p. 201)

Holland, Holyoak, Nisbett, and Thagard (1986) considered analogy as a second-order quasihomomorphism where the model of one real domain is considered as a model of another domain.

Clement (1988) restricted analogy only to the case where: (a) a subject, without provocation, refers to another situation B, where one or more features ordinarily assumed fixed in the original problem situation A, are different; (b) the subject indicates that certain structural or functional relationships may be equivalent in A and B; and (c) the related case B is described at approximately the same level of abstraction as A.

Eliot (1986) claimed that "research in many fields, including machine learning, cognitive psychology, and linguistics, does not make a clear distinction between the psychological phenomenon known as analogy and other types of problem-solving processes" (p. 17). The issue is whether such a distinction is either possible or necessary.

I do not believe that humans possess separate mechanisms for separate kinds of reasoning. I do believe that from a computational point of view, deduction, induction (generalization), and analogy are slightly different versions of a single uniform reasoning process. They differ in the outcome of the retrieval process and only with respect to this intermediate result and the correspondence between descriptions established during the mapping process, we can identify the reasoning process as deduction, induction, or analogy. In this way we can view the analogy case as the most general one with deduction and generalization at the two extremities—where the retrieved source and the target are related in a specific way, one of them happening to be a particular instance of the other.

Many researchers who model analogy separately suppose that, in the course of the reasoning process, an explicit decision to use analogy is made at the beginning, thus causing the application of the method of reasoning by analogy. For example, Wolstencroft (1989) stated explicitly that if we use one method in preference to any other one, we should have identified in advance that the chosen method will be the most likely to offer a solution, which is why he added an identification step to his model. In contrast with the above, I assume that typically the reasoning mechanism starts with its retrieval process and it is the result of the retrieval process which determines, at a later stage, the kind of reasoning used. Burstein and Collins (1988) and Collins and Michalski (1989), analyzing a set of protocols, also came to the conclusion that the kind of knowledge retrieved from memory drives the particular line of inference produced.

The present work is a part of a broader project aiming to elaborate and test the hypothesis about the uniformity of human reasoning. A uniform mechanism of human reasoning in a problem-solving task, called Associative Memory-Based Reasoning (AMBR), has been proposed (Kokinov, 1988b), and some experimental data supporting it has been obtained (Kokinov, 1990, 1992). As it is still in progress, in the current presentation I concentrate on the way AMBR models analogical reasoning, in spite of the fact that some of the considerations might be valid in other cases as well.

2. PSYCHOLOGICAL PHENOMENA TO BE MODELED

The general phenomenon to be modeled is that people do solve problems by analogy. This is, of course, well known from numerous experiments as well as from everyday life. We need, however, much more detailed information about they *way* people do it, which factors influence human performance and in what manner, and what kind of accompanying phenomena can be observed. There is a considerable shortage of psychological experiments that could provide answers to these questions, but there is some experimental data to be taken into account when modeling human analogical problem solving.

Analogical problem solving can be initiated by an explicit hint to use a particular case (provided by a teacher) as a source for analogy (Gick & Holyoak, 1980,1983), by a reasoner's explicit decision to try to solve a difficult problem by an (a priori unknown) analogy and generating (constructing) various sources by systematic transformations (Clement, 1988; Polya, 1954, 1957), or by spontaneous retrieval of a source from memory and noticing the analogy between this case and the target. In the present work only the last case is investigated: *the spontaneous use of analogy*.

It is a well-known experimental fact that people usually have difficulties retrieving spontaneously a source analog, especially an interdomain analog (Gick & Holyoak, 1980, 1983), and this is probably the main difficulty in human analogical problem solving. However, Holyoak and Koh (1987) demonstrated that spontaneous analogical transfer in fact occurs even between remote domains like the Radiation Problem (Duncker, 1945) and a lightbulb story. Experiments performed by various researchers (Centner & Landers, 1985; Gilovich, 1981; Holyoak & Koh, 1987; Ross 1984, 1987, 1989a, 1989b) demonstrated clearly that the main factor affecting the retrieval process is the semantic similarity between source and target (i.e., the number of shared features). Two different classifications of features as structural and superficial have been put forward in the relevant literature: sometimes the former are defined as causally related to possible solutions and the latter as features unrelated to any solutions, and sometimes the former are defined as n-ary predicates, especially the higher order ones, and the latter as unary first-order predicates. It was shown that superficial features (in both classifications) have considerably greater influence on the retrieval than the structural ones.

Gick and Holyoak (1983) demonstrated that the availability of a scheme (a more general and abstract description of a class of problems) aids in the retrieval of the corresponding source.

A study that is described in Section 2.2 demonstrates how different mental states influence the retrieving of an appropriate source and how these mental states can be affected.

A number of studies investigate human difficulties in establishing correct correspondences between the source and the target. It is particularly difficult to find correspondences between analogs from two different and remote domains. Even provided with the source and explicitly hinted, some subjects fail to use the analogy: about "25% of the subjects in experiments performed by Gick and Holyoak (1980, 1983) on the Duncker problem. It was demonstrated that the degree of structural consistency between source and target affects the ease of establishing such a correspondence but it was also shown that the similarity between the objects and relations involved in the analog situations is important as well (Centner & Toupin, 1986; Holyoak & Koh, 1987; Ross, 1987). In particular, it was demonstrated that crossmapping (similar objects playing different roles in the situations) impairs establishing a correct correspondence between source and target and that more similar relations are put in correspondence more easily.

In the following subsection I will briefly review an experimental replication of the results of Gick and Holyoak (1980) in a case study along whose lines computer modeling and simulation were done. Then, in Section 2.2,1 describe an experiment demonstrating priming effects on human analogical problem solving.

2.1. Difficulties in Human Analogical Problem Solving: A Case Study

Let us consider the following problem, further referred to as the "wooden vessel problem:"

imagine you are in a forest by a river and you want to boil an egg. You have only a knife, an axe, and a matchbox. You have no containers of any kind. You could cut a vessel of wood but it would burn out if placed in the fire. How would you boil your egg using this wooden vessel?

The subjects participating in the experiments have been asked to solve this problem. It appears to be a difficult one: the standard situation of the container being heated and conducting the heat to the water has to be rejected. Instead, the subjects have to develop an analogy with the process of heating water by means of an immersion heater for making tea in a glass, where the water receives the heat directly. Thus, possible solutions include heating the knife, the axe, or a stone in the fire and immersing it into the water in the vessel. Everyone has experience with immersion heaters (which are very popular in Bulgaria), so everyone can use this analogy potentially. However, even with the idea of an immersion heater in mind, it is hard to construct the analogy because, in contrast to many other analogies where a relation between the corresponding objects is transferred, in this case a new corresponding object has to be found in the target situation and only then the corresponding relations can be transferred. So this solution is of a highly creative nature.

Subjects have been tested in two different experimental conditions: (a) *control condition* when subjects have to solve the problem without any help, and (b) *hint condition* where they have been instructed to try to make an analogy with the case of using an immersion heater. As it can be seen from Table 5.1, very few subjects were able to solve the problem in the control condition (14%), while most of them were able to make the analogy when explicitly hinted, but with 35% still unable to construct the correspondence even then. A great number of

Table 5.1.
Results of Experiment I
control - hint: $\chi^2 = 62.17$ ($p<0.01$)

results/conditions	control	hint
success	14	104
failure	84	57
% success	14	65

these subjects wrote explicitly in their protocols that there were no immersion heaters or similar objects in the forest.

So two main difficulties have been encountered in human problem solving in this case: (a) recalling the "immersion heater situation," and (b) retrieving an object corresponding to the immersion heater in the target situation. It is obvious that both difficulties concern the retrieval mechanism, and the model has to explain them.

2.2. Priming Effects on Reasoning (Problem Solving)

In the experiment discussed above, since the "immersion heater situation" is well known to the subjects from their experience before the experiment, the *hint condition* results in only ignoring the retrieval process and immediately starting to seek a correspondence between the cases. In contrast to that, a *priming condition* would still rely on spontaneous retrieval of the "immersion heater situation" and noticing the similarities, but in addition to that the subjects' preliminary settings would be changed so that they could retrieve that source more easily. This is achieved by stimulating (activating) the source before presenting the target problem and in this way increasing its accessibility.

It must be noted that most priming effect experiments are performed with low-level tasks like item recognition, lexical decision, word completion, etc., while Kokinov (1990) explored the existence of priming effects in problem solving. The following reviews only part of these results (concerning only analogy) combined with the results obtained from some additional and more recent experimental sessions.

Table 5.2.
Results of Experiment II
control-near: χ' = 24.56 ($p<0.01$), control-far: χ^1 = 6.68

control-very far: χ' = 0.12 ($p>0.05$),
near · far: χ^* = 6.78 ($p<0.01$), near - very far: χ^2 = 18.55
($p<0.01$)
far - very far: χ^1 = 5.95 *[ρ* <0.05)

results/conditions	control	near	far	very far
success	14	71	35	7
failure	84	90	86	50
% success	14	44	29	12

Subjects have to solve a number of diverse problems including mathematical, physical, and common-sense ones in a mixed order. One of these problems is the target "wooden vessel problem" and another (prior to that one) is the priming problem: "how can you make tea in a glass." There are three different priming conditions: (a) the *near priming condition* where the priming problem is presented immediately before the target one, (b) *the far priming condition* where a single distractor problem (with a limit of 4 minutes for solving it) is given to the subjects between the priming and the target problems, and (c) the *far priming condition* where there are eight distractor problems (24 minutes) between the priming and the target ones. The priming effect is measured by the success/failure ratio rather than by reaction time because with such high-level tasks (as is problem solving) the reaction time depends on too many factors, it is difficult to measure and is therefore an unreliable parameter.

The results are shown in Table 5.2, and the differences between the four groups of subjects are found to be statistically significant applying the chi-square criterion. In this way it is demonstrated that: (a) there is a clear priming effect on analogical problem solving, (b) this effect decreases in the course of time, (c) it lasts for a certain period (at least 4 minutes) and, finally, (d) it disappears in less than 24 minutes. This can be illustrated by Figure 5.1.

All these results are to be explained by the model.

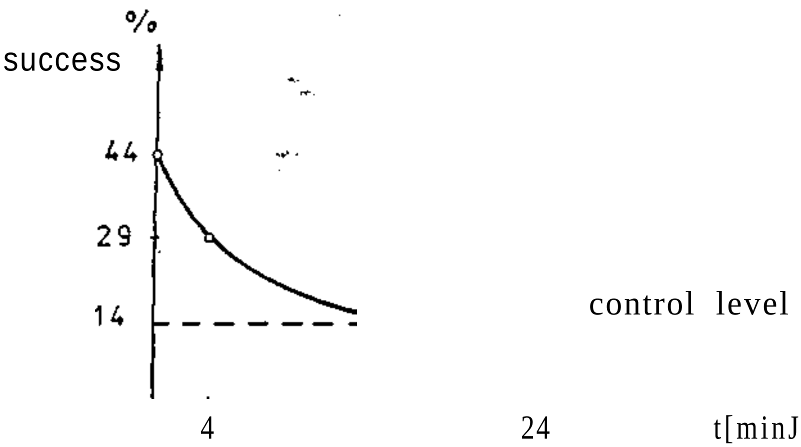


Figure 5.1.
The Decrease of the Priming Effect in the Course of Time (measured in minutes after the priming).

3. ASSOCIATIVE MEMORY-BASED REASONING (AMBR)

3.1. Dynamic Aspects of Structural, Semantic, and Pragmatic Constraints on Reasoning

Many researchers have suggested that various constraints should be imposed on the process of reasoning or on various subprocesses of that process. For example, Centner (1983) put an emphasis on structural constraints, whereas Kedar-Cabelli (1988) and Holyoak and Thagard (1989a) stressed pragmatic constraints. Most researchers take semantic constraints into account in their models to a certain extent. In recent years it has become clear that all three constraints are important at least at some steps in the reasoning process. So Centner (1989) included pragmatic constraints in her reasoning model (although only external to the mapping engine) while Holyoak and Thagard (1989b), Thagard, Holyoak, Nelson, and Gochfeld (1990) included structural constraints both on mapping and retrieval and built the ACME and ARCS models governed by all three types of constraints.

Holyoak and Thagard (1989b) gave clear definitions of structural, semantic, and pragmatic constraints. A structural constraint is the pressure to find and use an isomorphism between the source and the target description. A semantic constraint is the pressure to find and use correspondences between semantically similar elements of the descriptions. A pragmatic constraint is the pressure to find and use correspondences for pragmatically important elements of the descriptions. In the text that follows, the pragmatic constraint is considered in more detail, and after that, its relations with the semantic and structural constraints are briefly discussed.

3.1.1. Context and Relevance. The key issue in the pragmatic aspect is the way in which important (relevant) elements are defined. Relevance is always defined with respect to a particular context, hence, two questions arise: what is considered as a context, and what are the criteria for determining the relevance?

Typically only the *problem context* is taken into consideration (i.e., the relevance of an element is defined with respect to the whole problem description (Anderson, 1983; Mitchell & Hofstadter, 1990). In some models (Eskridge, this volume; Seifert, 1994) the contextual goal of the reasoner (e.g., problem solving, learning, explaining, etc.) is also taken into account. I would like, however, to consider the *whole problem-solving context* (i.e., the entire real-world situation) within which a solution of a problem is being searched. There are generally two parts of this problem-solving context:

- the *external context,* consisting of the reasoner's representations of the currently perceived part of the environment which is not necessarily related to the problem situation (the reasoner cannot be isolated from the environment);

- the *internal context,* which encompasses the reasoner's current state of mind, including the currently active goals, knowledge, thoughts, etc. (the reasoner never commences the problem-solving process with a "blank" mind).

The problem description is included either in the internal or in the external context, or possibly in both of them.

3.1.2. Causal and Associative Relevance. Relevance can be defined in different ways, depending on the choice of the context and the criteria.

Typically, relevance is defined with respect to the goal of the reasoner (which is part of the problem context), and the criterion for relevance of an element is whether a causal chain connecting that element with the goal exists (Thagard et a!., 1990). I call this *causal relevance.*

Another criterion for relevance with respect to the whole problem-solving context can be the degree of connectivity of the element in question with all other elements of that context. This criterion is based on the reasoner's implicit assumption that things that happen simultaneously are probably causally related (which forces a tendency to link co-occurring events or features). This is not always true, but it provides a criterion for relevance that is both dynamic and easy to test. I call this *associative relevance.*

3.1.3. Why Do We Need both types of Relevance? In an artificial situation where only the problem description forms the context (e.g., where the list of all possible actions and/or instruments is provided for in the problem description) it is possible, at least theoretically, to test the causal relevance of each action or instrument. In a realistic context, however, the reasoner has to elicit the possible actions from memory and the possible instruments from the real-world environment. Thus, it is impossible simply to test all the possibilities because explicit knowledge about most (or all) of them will be unavailable a priori.

People, however, have an intuitive idea of the important aspects of a situation even before there is any possibility of formal analysis of the situation and sometimes even before the goals are defined or made explicit. In other words, the reasoner will know that a particular element is somehow connected to other pieces of knowledge, presently considered as relevant, without being able to report the exact nature of these connections or a particular path followed. In this way associative relevance can be considered as a preliminary and approximate estimation of the relevance of all memory elements to the whole context. Only the ones estimated as most relevant are eventually tested for their causal relevance (i.e., for their particular relevance to the goal of the reasoner).

Let us recall some famous examples where the particular external context has reportedly played a crucial role in human reasoning:

- Archimedes discovered his law in the bathroom seeing the water overflowing the bath when he entered it.

- Seeing an apple falling from a tree gave Newton inspiration for his theory of gravity.
- John Atanassov (one of the inventors of digital computers) decided to use electronic tubes for his computer when he saw a row of bottles in a bar.

In all those cases it was the particular *external context* which made the corresponding memory elements associatively relevant and only then a more formal analysis elucidated the causal relations (if any) between the perceived event and the goal of the reasoners. Formal analysis of *all* events perceived from the environment was definitely *not* performed; only those "felt" to be relevant were formally analyzed.

On the other hand, the priming effects demonstrated in our experiments manifest the influence of the particular *internal context* on the associative relevance of memory elements and, thus, on the line of reasoning.

3.1.4. Differences between Causal and Associative Relevance. The two types of relevance considered above seem to have very different properties (Table 5.3). For example, causal relevance appears to be more static since it depends on the problem goal and is thus highly connected to the problem itself (i.e., whenever we present one and the same problem, the same elements are expected to be considered important as they will always be connected to the goal of the problem)[2]. On the contrary, associative relevance is highly dynamic and variable because of the continuously developing external and internal contexts (note that it is impossible to replicate any particular context). This throws light on the causes for the variability of human problem-solving behavior and, in particular, the priming effects demonstrated in Section 2.2.

The causal relevance criterion can be used to determine whether or not a path to the goal exists, but it is difficult to define a *measure* of causal relevance. Although it is possible to obtain such a measure by selecting certain characteristics of the path (e.g., its length) for evaluation, in this way an *absolute* measure would be produced. It is more natural, however, to consider relevances only in relative terms (one entity being more relevant than another). Further, a *relative measure* implies ordering all the relevance measures and that would be impossible since this requires computing the causal paths for all elements in advance which is unrealistic. That is why causal relevance is better defined to be of type "all or none." On the other hand, associative relevance is by definition *graded* because it is clear that all elements are somehow related to each other, so it is the degree of relevance that matters. Moreover, there exists an efficient mechanism for computing associative relevance for all elements at once. This associative relevance is a kind of distributed representation of the situation in human memory showing the pragmatic importance of each memory element.

[2] However, since the finding of a causal path can depend on its associative relevance, causal relevance can also be considered as dynamic and context-dependent.

Table 5.3.
Different Properties of Causal and Associative Relevance

Relevance type	Depends on	Type	Temporal aspects
Causal	goal	all/none	static
Associative	problem-solving context	graded	dynamic

These differences in the properties of causal and associative relevance lead us to propose different mechanisms for their computation. Causal relevance in AMBR is computed by a marker-passing mechanism (described in Section 4.7) analyzing the reasoner's goals and traversing causal relations, while associative relevance is computed by the associative mechanism described in Section 4.5 which is a form of spreading activation. Thus, associative relevance is measured by the degree of activation of the corresponding element.

5.7.5. ***Dynamic Aspects*** *of Semantic Similarity.* The nature of the semantic constraint depends on the definition of similarity. Semantic similarity can be defined in the terms of the entities' representation and of their location in memory organization.

A classical approach to semantic similarity is to measure it by the degree of overlap between feature representations of entities (Stanfill & Waltz, 1986; Tversky, 1977).

A second approach is to measure similarity between entities by the distance between them in the memory organization (i.e., entities within the same neighborhood are more similar than those far away in the memory organization). Schank (1982) proposes episodes in memory to be organized in a way that allows episodes represented by very different features to be within the same neighborhood (called ΓOP) if they share some.more abstract relationships between goals and plans. Thagard et al. (1990) define two relations to be semantically similar if they are identical, synonyms, hyponyms (are of the same kind), or meronyms (are parts of the same whole), that is, if they are immediate associates. Objects are semantically empty and their similarity is determined on the basis of their participation in similar relations.

In AMBR, two entities (either objects or relations) can be considered as similar if a common point of view on them can be found (i.e., if a common superclass at any level can be found)[1]. Moreover, the degree of semantic

[1] Two entities are considered to be similar also when they correspond to two points of view on the same thing (i.e., both of them represent one and the same object or concept in the world) or if a mapping between their descriptions can be established (which is dynamically computed at the moment of comparison).

similarity corresponds to the associative relevance of this common superclass found. Therefore, an a priori restriction to immediate superclasses is unnecessary when computing the similarity between entities; instead, the search can potentially be extended to superclasses at any level, relying on the relevance factors to prevent it from becoming exhaustive (more details can be found in Section 5,2).

Holyoak and Thagard (1989b) and Thagard et al. (1990) consider the pragmatic and semantic constraints as independent inputs to their constraint satisfaction machine competing later with each other. In contrast to that, I suppose that the computation of semantic similarity cannot be done independently, without using information about the associative relevance of the pieces of knowledge in memory. Thus, two entities can be considered as dissimilar regardless of their potential similarity if the respective aspect is not relevant to the context. For example, two cars (mine and that of somebody else) can be considered as dissimilar (although being instances of the same class) in the context of owners, possession, properties, etc., whereas my car and my house will be considered as similar in the same context. This makes similarity itself both context sensitive and having a dynamic nature.

3.1.6. Dynamic Aspects of the Structural Constraint. Since exact isomorphisms cannot usually be found, certain priorities have to be assigned to particular elements (e.g., Gentner [1983] claimed that higher order relations have to have higher priority [the systematicity principle]). In our model each Structural pressure has its own particular weight, depending on the associative relevance of the corresponding elements (i.e., it is context-dependent and, therefore, dynamic). In particular, when the causal relations or other higher-order relations are highly relevant to the context, the systematicity principle will be in force. This treatment of interaction between structural and pragmatic constraints is similar to that of Holyoak and Thagard (1989b) except for the context-dependent nature of relevance in our model.

While Gentner (1983) embedded a strong semantic constraint ,µι the structural one allowing only identical relations to be mapped, Holyoak and Thagard (1989b) considered semantic and structural constraints as completely independent and allow any relations to be mapped independently on their semantic dissimilarity. I take an intermediate position: a structural constraint can start only from semantically similar entities (i.e., if two propositions, relations, or objects are already evaluated as similar, they will impose structural restrictions on their arguments, otherwise no restrictions are presumed). That is why the structural constraint depends on semantic similarity and therefore, once again, on pragmatics. This adds to its context-dependent nature.

So the pragmatic constraint plays a dominant role in our theory (i.e., all other constraints are computed on the basis of the associative relevance factors and therefore are rendered context-dependent and dynamic).

3.2. Parallel Running and Interaction Between Components of AMBR

AMBR has been proposed as a computational model of human reasoning in a problem-solving task (Kokinov, I988b, 1990; Kokinov & Nikolov, 1989). It consists of the following components: *retrieval, mapping, transfer, evaluation,* and *learning.* These components are widely used for describing analogy (Hall, 1989; Wolstencroft, 1989), but most models consider them as sequential steps in the reasoning process and even try to deal with them separately.

Each of the components of AMBR will be discussed in greater detail in Section 5, only their objectives will be formulated here.

In contrast with the typical case where the aim of the retrieval process is to select one piece of knowledge for further manipulation, *the aim of the retrieval process* in AMBR is to compute *the associative relevance of each piece of knowledge.* Thus, as a result of the retrieval process we have the associative relevance factor of each entity and this factor plays an important role in all other processes. As a side effect, the most relevant piece of knowledge (called focus) is determined and it can serve as a potential source of analogy.

The aim of the mapping process is to establish a correspondence between two descriptions. In the case of AMBR, these are the focus and the input (or goal) structures. As the focus changes over time a number of different mappings can be initiated to run in parallel.

The objective of the transfer process is to extend a given correspondence between two descriptions in order to add elements (inferences) to the target description. The latter correspond to elements in the source description. In this way knowledge is transferred from the old situation to the new one.

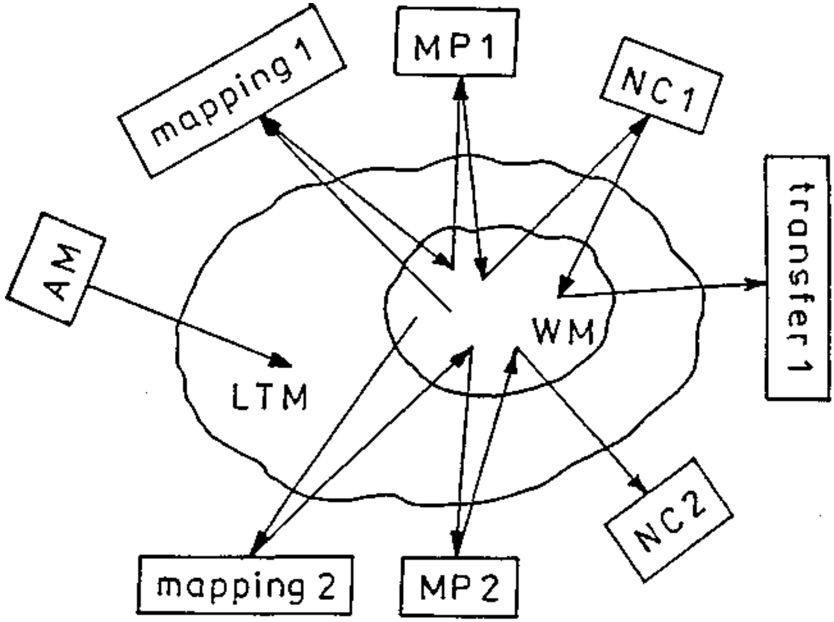
The aim of the evaluation process is to estimate the consistency, plausibility, validity, causal relevance, and applicability of the inferences.

Finally, *the objective of the learning process* is to modify the reasoning system itself in a way that improves its later problem-solving behavior. This involves storing the new structures together with the inferences, making generalizations (inducing problem schemas) if possible, storing problem-solving traces (failures, successful steps, etc.), and adjusting the links to enable better retrieval in the future.

In contrast with many other models (including Centner, 1988, 1989; Holyoak & Thagard, 1989a) in AMBR these components are considered as processes running in parallel and communicating through a global "database" (the LTM of the architecture) rather than as sequential steps in the reasoning process (Figure 5.2)/

[4] Actually, the processes running in parallel are also part of LTM. The mechanisms allowing this parallelism are described in Srction 4.

Figure 5.2.
Parallel Running of Various Components of Two Concurrent Reasoning Processes, where *AM* Stands for Associative Mechanism, *MP* for Marker-Passing Mechanism, and *NC* for Node Constructor (AM performs the retrieval process whereas MP and NC are components of the mapping process).



3.2.1. Why is this Parallelism Necessary? The continuous development of both the external and the internal contexts over time requires that the process of retrieval is running continuously and in parallel with all other processes, changing the relevance factors of the entities and thus influencing all other processes.

Learning also has to run in parallel with the other processes in order to be able to store intermediate results, maps, failures, etc. Evaluations should be made in parallel to other processes thus guiding the reasoning process.

People often perform several complex actions simultaneously (e.g., driving a car and talking, cooking and planning the activities for the next day, lecturing and trying to develop the opponents' arguments). This would require several mapping and transfer processes running in parallel, establishing correspondences between different structures.

Although I am not aware of any formal experimental study of the possibility of several mappings running in parallel, solving one and the same problem or different ones, there is some interesting introspective evidence that makes such an assumption plausible. Hadamard (1945) studied carefully several reports provided by well-known mathematicians on how they came to their interesting inventions and also interviewed a number of his contemporaries. He discovered that often insight (spontaneously seeing the solution of a hard problem) occurred while researchers were thinking of completely unrelated things. So he suggested an explanation that people are actually reasoning in parallel on various problems without being aware of that fact (only one of these reasoning processes being at the conscious level) and when a good "aesthetic" result is obtained by one of the other reasoning processes, this result becomes consciously available. This explanation has, of course, a speculative character and has never been tested but it is nevertheless interesting and stimulated me additionally to propose such a parallel architecture.

In short, parallelism in the architecture is introduced both in order to support:

1. mutual interaction between the components of a reasoning process, and
2. concurrency in the running of several reasoning processes each possibly associated with a different task.

4. A HYBRID COGNITIVE ARCHITECTURE

A theory *of cognitive architecture* is a theory of the basic principles of operation built into the cognitive system (Anderson, 1983). The cognitive architecture is an integrative explanation of cognition that comprises a unified description of mental representation, memory structures, and processing mechanisms. In the recent years, several proposals for cognitive architectures have been made, for example, ACT* (Anderson, 1983), The Society of Mind (Minsky, 1986), Soar (Laird, Newell, & Rosenbloom, 1987; Lewis et al., 1990; Newell, 1990; Rosenbloom, Laird, Newell, & McCarl. 1991), PUPS (Anderson & Thompson, 1989), and PRODIGY (Carbonell et al., 1990).

Studying human cognition I have been led by the assumption that it is not possible to build an adequate model of an isolated cognitive phenomenon. Cognitive processes are too complex and interrelated to be modeled separately, and I believe that it is necessary to have a *cognitive architecture* on the basis of which different models of different phenomena can be proposed.

Consequently, a cognitive architecture was put forth by Barnev and Kokinov (1987) and Kokinov (1988a), which was developed further in Kokinov (1989) and Kokinov and Nikolov (1989) and in the present chapter. On the basis of that architecture, a model of human recalling and forgetting (Kokinov, 1989) as well as a model of human reasoning (Kokinov, 1988b; Kokinov & Nikolov, 1989) have been proposed.

The cognitive architecture described here is a hybrid one. It combines the symbolic approach (a frame-like representation system with parallel running symbolic processes and a marker-passing mechanism) and the connectionist approach (a localist connectionist network with an associative mechanism). The symbolic aspect of the architecture performs the reasoning proper whereas the connectionist aspect makes it effective, context-dependent, and dynamic.

4.1. Dualistic Representation

As we have seen so far at least two aspects of the world have to be represented in the reasoner's mind: (a) knowledge about the world (concepts, objects, events, plans, etc.), and (b) their associative relevance with respect to the particular context. These two aspects are orthogonal.

The proposed architecture reflects both aspects. Concepts, objects, situations, plans, actions, etc.. are naturally represented by corresponding descriptions (frame-like symbolic structures), whereas their associative relevance is represented by the level of activation (a numeric value) of these descriptions.

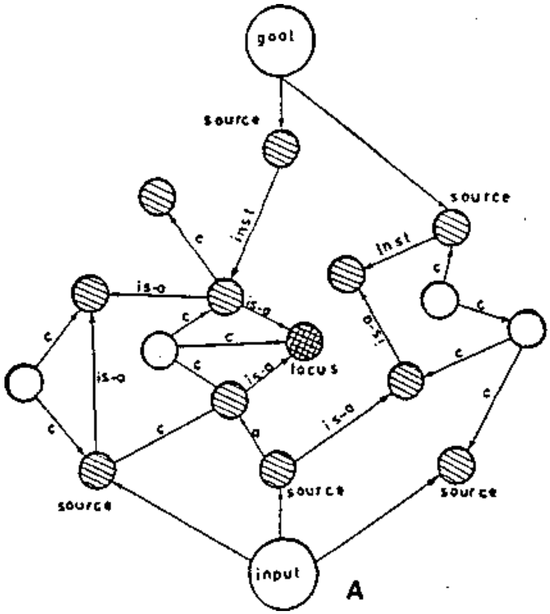


Figure 5.3a.
Long-term memory as a network of frames and semantic relations between them. An example of a particular memory state is depicted, where *is-a* stands for is-a link, mst for *instance-of, c* for *c-coref,* and a for *a-link,* shaded (hatched) nodes stand for activated elements of LTM, source nodes are

We can consider the frames as nodes and their slots as links between the nodes (actually a frame in our representation scheme is nothing more than a bunch of highly structured and named links). In this way the *long-term memory (LTM)* is considered as a network of nodes and links (where nodes correspond to concepts, objects, events, actions, situations, etc., and links to semantic relations and arbitrary associations). Ascribing weights to the links and activation to the nodes, and abstracting from the semantic interpretation of the links, we can think of the LTM as a large localist connectionist network.

In this way each link and node in the network has a *dual interpretation:* one within the symbolic representation (Figure 5.3a), and one within the connectionist representation (Figure 5.3b).

So, each link: (a) has a semantic label and fulfills different roles in the symbolic representation scheme, and (b) has an ascribed weight and serves to convey activation to neighboring nodes within the connectionist network.

Each node corresponds to: (a) a frame!ike description in the symbolic representation scheme, and (b) a simple unit in the connecitonist network with an activation level corresponding to the *degree of associative relevance* of that conceptual description.

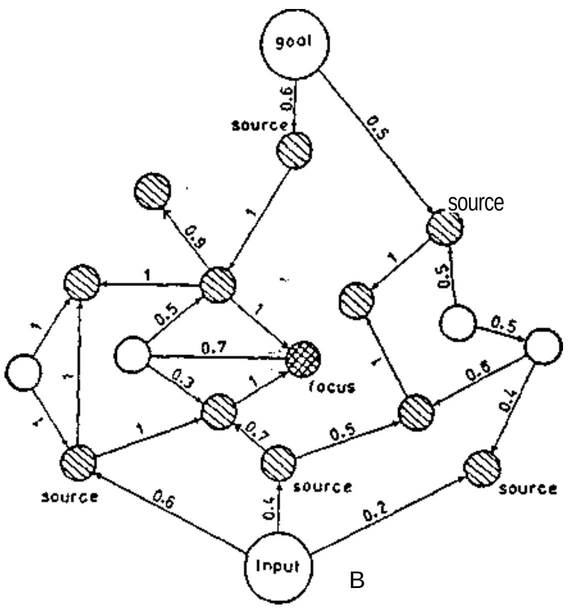


Figure 5.3b.
Long-term memory as a connectionist network.

sources of activation and *focus* is the most active node. For a detailed explanation of the links and nodes in both aspects of the architecture see the text in the following subsections.

4.2. Nodes as Processors

There is, however, another dualism. In addition to the interpretation of nodes as *representational elements,* it is also possible to consider them as *processing elements* (Table 5.4).

From the connectionist perspective, they are simple numeric processors calculating the activation values and outputs of the nodes on the basis of their input values and current activity running in parallel in a discrete synchronous manner in order to simulate the continuous process of spreading activation.

From the symbolic perspective, they are specialized symbolic processors able to receive and send markers (pointers to other, possibly nonneighboring nodes), to differentiate links with different labels, and possibly to perform specific hard-wired programs corresponding to some actions of the reasoner. Symbolic processors run in parallel in an asynchronous manner, each at its own individual speed. The reasoner has a number of symbolic processors with one and the same hard-wired program (e.g., mapping, marker passing). This allows for performing several different mapping processes in parallel. The exact limit of the number of processors of one and the same type has to be a subject of experimental estimation for each particular set of processors.

4.3. Interaction between the Symbolic and Connectionist Aspects of the Architecture

The interaction between the symbolic and connectionist aspects is realized through the close relations existing between the symbolic and connectionist processors corresponding to a given node.

The work of the symbolic processor influences the work of its connectionist counterpart in the following way. "Good" results obtained by a symbolic processor increase its own activation level, and "bad" results decrease it. For example, if a node has received markers from two different origins, this raises its activation; on the contrary, if a symbolic processor fails in doing its job for some reason then its activation is suppressed.

Table 5.4.
Nodes as Processors

Perspective	Computation	Parallelism
Symbolic	symbol processing	asynchronous, individual speed
Connectionist	numeric computation	synchronous, instantaneous jumps

On the other hand, the connectionist processors influence the work of their symbolic counterparts as well. In general, the connectionist processor can be considered as an energy supply for the symbolic one (i.e., the higher the activation level of the connectionist processor, the more productive the symbolic processor).

Let us first consider the case when the node represents a reasoner's possible action (including a mental one). If the activation level of that node obtained by the connectionist processor is above its threshold, then the symbolic processor will be started and it will run with a rate proportional to this activity. In this way a set of symbolic processes runs in parallel and with different rates at each particular moment. These processes can communicate with each other through the links between them. Each processor has, however, a sensitivity threshold (i.e., the minimum level of activation that another node has to possess in order to be able to pass on the markers sent by this processor) associated with it which limits its communication abilities. This threshold can be absolute or relative and may depend on the activation level of that node. In this way only part of the nodes in WM are available for the corresponding symbolic process.

If now a node represents a concept, object, or some other declarative knowledge, then the greater its activity, the more processors will be able to use it. On the contrary, if, for example, the node is inactive, then it will be inaccessible for all processors.

In this way the connectionist aspect of the architecture continuously "restructures" the knowledge base of the reasoner represented by the symbolic aspect thus controlling the set of possible inferences at any moment, it makes some nodes more accessible and others completely inaccessible, thereby assigning priorities, restricting the search, etc. This makes the knowledge base dynamic and context-dependent.

4.4. Localist Connectionist Network

Within this aspect of the architecture the *long-term memory (LTM)* is considered as a large localist connectionist network (Figure 5.3b). The nodes have variable levels of activation ($0 \leq a_i(t) < 1$). All nodes whose level of activation is above their threshold t, *form the working memory (WM).* These are the only nodes accessible by the running symbolic processes. There are also nodes which are sources of activation—these are the input and the goal nodes (i.e., nodes corresponding to entities (in the external world) being perceived at the moment and nodes corresponding to the reasoner's goals). The *source nodes* emit activity continuously (i.e., they have a constant level of activation for the period of time they are on the input or goal list). The total amount of activation emitted by all goal and input nodes is limited and distributed among them proportionally to the weights of the links connecting them to the goal and input nodes, respectively. (Perception has its own focus and goals have their priorities.) The node in WM (excluding the source nodes) with maximum level of activation at a given moment is called *the focus* (of attention). A change of the focus may take place only after another node has been the most active one for a *sufficiently long period of time* (a temporal threshold). The activation of the focus node does not decay but it is not a source of activation.

The links in LTM are *excitatory* only. They are directed and have weights ($0 < w_{ij} < 1$) ascribed to them that correspond to their strengths. Some symbolic processes (e.g., the mapping one) may establish additional temporary links which can be *both excitatory and inhibitory* ($-1 < w_{ij} \leq 1$).

The weights of the links as well as the thresholds of the nodes are subject to changing and learning. Only excitatory temporary links can become permanent, that it, become part of LTM.

4.5. Associative Mechanism

The associative mechanism is responsible for changing the activity of the nodes thus changing the state of WM. It is a form of relaxation search. *Relaxation search* is a continuous process which serves two purposes: determining associative relevance, and performing constraint satisfaction.

To determine the associative relevance of each piece of knowledge in LTM, relaxation search as a form of spreading activation (Anderson, 1984) is performed on an excitatory network with positive (only) links reflecting mutual support between nodes regarding their relevance to a situation.

Constraint satisfaction is used to find the best mapping between two descriptions (see Section 5.2) or a single interpretation of the input. In this case, inhibitory as well as excitatory links are temporarily added to the existing ones and relaxation search is performed on the resulting constraint network (including LTM).

The detailed description of the relaxation search will be given in connectionist terms (Rumelhart, McClelland, & the PDF Research Group, 1986). It is simulated by a discrete synchronous process; at each moment t every node has some activity $a_i(t)$ and passes some output—$o_i(t)$—to its neighbors. I suppose that

$$o_i(t) = \begin{cases} 0 & \text{if } a_i(t) < \tau_i \\ p \cdot a_i(t) & \text{otherwise,} \end{cases}$$

where $0 < p < 1$. Each node receives activity from all of its neighbors and the total input from them is the weighted sum of all their outputs $net_i(t) = \sum_j \omega_{ij} o_j(t)$, where ω_{ij} are w_{ij} normalized at the time of computation of $net_i(t)$ so that $\sum_i \omega_{ij} = 1$ (for all weights of links leaving an arbitrary node n_j).

There is a decay process as well, which exponentially decreases the activity of all nodes in WM with the exception of the focus, **the input and the goal nodes.** Finally, the sum total of activity in a node is:

$$sum_i(t) = \begin{cases} a_i(t) + net_i(t) & \text{if } n_i \text{ is the focus} \\ \tau \cdot a_i(t) + net_i(t) & \text{otherwise,} \end{cases}$$

where τ is the decay rate, $0<\tau<1$. (The parameter τ corresponds to the loss of energy (activity), and can be different in different reasoners.) The activation level of the node at the next moment of time is computed from this sum in the following way:

$$a_i(t+1) = \begin{cases} 0 & \text{if } sum_i(t) \le t_i \\ 1 - (t_i/sum_i(t)) & \text{otherwise,} \end{cases}$$

where t_i is the activation threshold for node n_i (Figure 5.4). This threshold determines the excitability of the node: nodes with low thresholds are easily accessible by processes or correspond to easily triggered processes.

4.6. The Symbolic Representation Scheme

A frame-like representation scheme (Minsky, 1975) is used for several reasons: (a) the possibility to have several different frames for a single object or concept reflecting different points of view, and (b) the integration of declarative and procedural knowledge in common structures. Descriptions correspond to con-

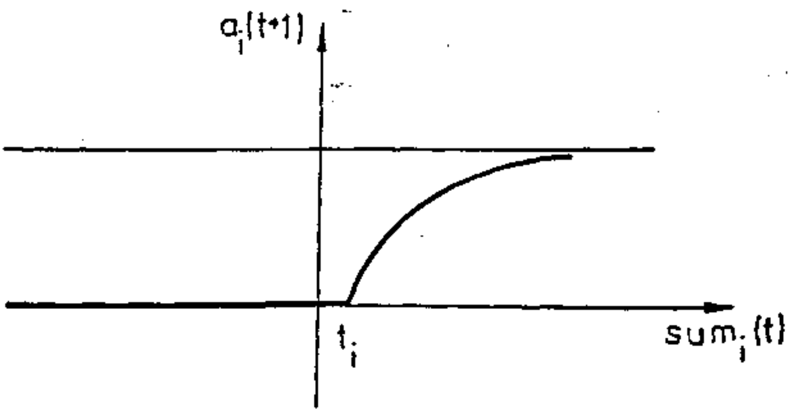


Figure 5.4.
The Activation Function of the Connectionist Network.

cepts, objects, events, situations, propositions, plans, actions, etc. A detailed description of the representation scheme can be found in (Kokinov, 1989). Only some of the more important aspects of it will be outlined here. The *is-a* and *instance-of* slots define the concept as a specialization of or a particular instance of a class description. In such cases, the same kind of links are provided from each slot to its superslot in order to establish the slot correspondence (neither the name of the slot nor its position can establish this correspondence).

The *c-coref* (short for conceptual coreference) slot points to other conceptual descriptions of the same entity (i.e., each two descriptions linked to each other by a c-coref link refer to one and the same entity in the world). This allows for multiple descriptions of one and the same object, concept, situation, etc. The *c-coref facet of a* slot is used to represent a *c-coref* link between a part of the description and another description or part of it. This facet replaces the value and range facets, provides a way of specialization of an ancestor slot and also makes it possible for two or more frames to share some information and in this way to build a frame array (Minsky, 1975, 1986). A detailed description of all these properties can be found in (Kokinov, 1989).

The *a-links* (short for "associative links") represent arbitrary associations corresponding to any type of co-occurrence or other vague semantic relations. They are not recognized by the symbolic processors and are used only by the connectionist aspect of the architecture. For example, such links are built between characteristic features often found together, between two events that have occurred within a short period of time, from class descriptions to their elements or subclasses, etc. These links enable the computation of associative relevance of the particular pieces of knowledge linked.

Besides these common slots, each frame may have an unlimited number of special slots. They can be of three types: aspects, relations, and actions. The aspect slots of a frame correspond to structural or functional parts of a concept (e.g., a handle of a cup), abstract aspects of a concept (e.g., a scientist's research area), or roles in a relation or an action (e.g., the agent, the object, and the recipient in an action of giving). The relation slots correspond to relationships between different aspects of the same concept or between different concepts or their aspects (e.g., in a frame describing a situation, aspects can correspond to objects, whereas relations can correspond to their attributes or relations between them). Action slots correspond to pieces of behavioral knowledge attached to the concept or situation (e.g., how to act in such a situation). This may be viewed as the hard-wired knowledge of the corresponding symbolic processor.

4.7. Marker-Passing Mechanism

4.7.1. The Role of the Marker-Passing Mechanism. In contrast to the associative mechanism which performs an undirected and unspecified search and produces only numeric reports, the marker-passing mechanism performs a specialized search traversing only the links of a specified type (e.g., either is-a links, instance-of links, c-coref links, or a certain combination of these types) and keeps track of the path followed. It is actually a mechanism for directed parallel retrieval. This mechanism is used for establishing the causal relevance of a particular element as well as for establishing the semantic similarity between various elements during the mapping process.

The mechanism has two different versions. In the first case, it starts from two nodes in the network and looks for paths beginning at these nodes, consisting of the links of the given type(s) and ending at one and the same node. In the most typical case, the links involved are traversed possibly starting with one *instance-of link* (which may be omitted) and then continuing with an unspecified number of *is-a* or *c-coref* links in an unspecified order. Such a search will answer the question whether two given descriptions corresponding to the two starting nodes have a common super- or metaclass (i.e., they are subclasses or elements of one and the same class); this is used during the mapping process as well as during the evaluation process.

There is a second, slightly different version of the marker-passing mechanism where a correspondence of the same type is sought between the elements of two given sets of nodes. This mechanism is used during the mapping process as well.

4.7.2. The Marker-Passing Mechanism itself. In the first case, it marks the two starting nodes with two different markers (e.g., A and B), which are then passed through the specified links, marking all nodes on the path along which the corresponding marker is being passed. All nodes marked by both markers are reported together with their activations.

In the second case, there are two sets of markers (e.g., Al, A2, .. and Bl, B2, ..) used for the two initial sets of nodes, respectively, each node receiving its own unique marker. The markers are passed through the network in the same way as in the first case. Then the nodes marked by markers both of the A/ and the B/ type (i.e., the crossroad nodes) are reported together with their activations. Note that if, for example, a node is marked by markers AI, B2, and B3, then correspondences are established between Element 1 of the first set and Elements 2 and 3 of the second set (Figure 5.5).

The way the markers spread throughout the network is highly influenced by the activation of the nodes. Markers do not pass through inactive nodes or through nodes whose activation is below the sensitivity threshold of the originating marker-passing processor (since several marker-passing processes can run in parallel, markers carry information identifying the specific processor initiating the marker-passing process and its sensitivity level). Moreover, markers pass faster through more active nodes. This is due both to the fact that the corresponding symbolic processors run at higher rates and to the fact that they carry markers over to their neighbors in an order corresponding to the order of their activation levels.

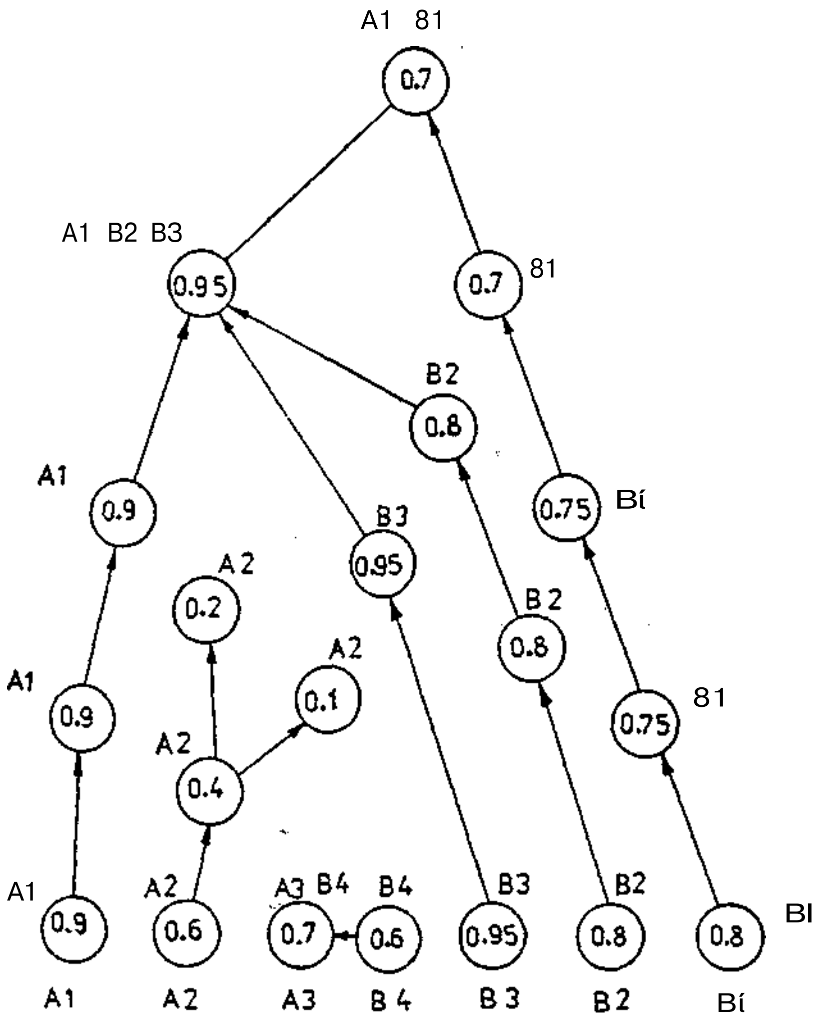


Figure 5.5.
An Example of the Work of the Marker-Passing Mechanism. Only marked active nodes at a given moment are shown with their markers. The reports will be (A1 B3 0.95), (A1 B2 0.95), (A1 B1 0.7), (A3 B4 0.7). Note that different markers arrive at different moments of time, and reports are produced as they become available.

4.8. Manipulating the Environment and Goals

The considered model is one of an active reasoner that not only reacts to the input provided by the environment but has her own goals and is able to change the environment.

Thus, for example, the reasoning process itself can put forward goals which in their turn will influence it. This is performed by putting a description (i.e., a node having the appropriate content) on the goal list.

On the other hand, it is possible that the reasoning process needs additional information not present in the environment at the moment. This will cause the activation of an action process that will have a certain effect on the environment; this interaction will probably lead to a change in the input to the reasoning system. As a result of that, the reasoner will perceive a different environment or there will be a reordering of the input nodes with respect to their activity (which depends on the characteristics of the incoming signal); in effect there will probably be a shift in the reasoner's attention.

Let us recall that the ability of the reasoner to enrich the goal structures stored in WM is limited by the constraint on the total activity of all the goal nodes. That is the reason why the reasoner widely uses the environment (via actions and perception) as "external memory" to increase her reasoning capacity. This is achieved by manipulating the environment appropriately, for example, by adding things to the environment (in the typical case of a human reasoner—by writing down notes about some of the goals), and by perceiving the results of this manipulation at a later stage (in the case of a human reasoner—by reading the notes). Thus, an external goal structure accessible by perception is used instead of, or in addition to, an internal one. Although there are restrictions on the input list as well (it seems plausible [Miller, 1956] that the entries on the combined goal and input lists after the seventh one receive too little activation), this still extends the reasoner's capacity enormously since her attention can be focused on different parts of the environment at different moments.

It is also plausible to create a mental process that simulates such an interaction with the environment (Rumelhart, 1989) by directly manipulating the input list (i.e., this is in fact imagining an interaction with the world). However, the internal resource limitations will reemerge in this case.

4.9. Learning

There are two main ways of learning in this architecture: (a) by constructing nodes, and (b) by adjusting links.

Node construction actually consists of two steps: (a) generation of a new node and (b) adjustment and interpretation of its links. It is possible to start with an uninterpreted link (an a-link) and, after it has been strengthened to its maximum weight of 1, to give it some interpretation (usually as a part-of relation) and at this moment to make it a part of the description frame corresponding to this node (i.e., one of its slots).

There are also some processes which directly construct new temporary nodes (e.g., the node constructor process discussed in Section 5.2) some of which can later become permanent.

Link weights are changed according to a sort of competitive learning. The only node whose links weights are changed is the focus. The strengthening of the links connecting the focus to its neighbors is proportional to the activities of the neighbors. This is done by computing the mean activity of all neighboring nodes and changing the links beginning at the focus according to the formula:

$$w_{focus,i} = \beta(a_i(t) - a_{mean}(t)).$$

The normalization of the weights of the outcoming links (so that $\Sigma_i \omega_{f_i} = 1$) is done each time they are used (i.e., the actual link weights are dynamically computed). When the weight of a link reaches the maximum level of 1, then it is not changed any more.

New associative links *(a-links)* are always built between the input nodes, the goal nodes, and the focus, as well as between the input nodes and between the goal nodes themselves.

5. COMPUTATIONAL MODEL OF ANALOGICAL PROBLEM SOLVING

This section treats in greater detail the process components of analogical reasoning, how the mechanisms of the proposed cognitive architecture contribute to the model of reasoning by analogy, and how the model explains the empirical facts.

5.1. The Retrieval Process

> To use an analogy, gaining access to LTM, is a bit like fishing: the learner can bait the hook—that is, set up the working memory probe—as he or she chooses, but once the line is thrown into the water it is impossible to predict exactly which fish will bite. (Centner, 1989, p. 231)

5.1.1. Mechanisms of Retrieval. There are two mechanisms of retrieval in AMBR: *automatic* and *strategic.*

Automatic retrieval is the process responsible for keeping the memory state of the reasoner in correspondence with the current context. It follows the development of the context and reflects its changes by continuously recomputing the associative relevance of all memory elements, to be used by the other symbolic processes. It is performed by a process of automatic spreading activation with the underlying assumption that the activity of a node reflects its associative relevance to the whole problem-solving context (including input from the environment, goals, current ongoing reasoning processes, etc.). Goal and input nodes are sources of activation and all currently active nodes (the elements of WM) keep for a while some residual activation. Automatic spreading activation is performed by the associative mechanism described in Section 4.5, which is a form of relaxation search in a connectionist network.

Problem solving starts when a description of the target problem is constructed,[1] put on the goal list, and some of the concepts used in its formulation as well as some unrelated objects from the environment are on the input list. As automatic retrieval is running continuously, associative relevances are then recomputed thus reflecting the new state of the reasoner. Each symbolic process uses this information in its own way. For example, mapping uses the most active node (the focus) as a source of analogy as well as the relevance factors of all other nodes in the process of establishing semantic and structural correspondences between the source and the target descriptions (see Section 5.2).

So in order to retrieve a source for analogy the reasoner has "to bait the hook," that is, to set up the goal list (to put the target description on it), possibly to concentrate (by looking, touching, etc.) on particular objects in the environment (i.e., to put the corresponding nodes on the input list), and to wait for the result of the associative mechanism: the established focus can be considered as a possible source for analogy. If the mapping process fails to establish a correspondence between this source and the target or if the evaluation process finds the established mapping inadequate, then this focus is deactivated and the associative mechanism establishes another focus which is considered as another possible source for analogy.

The success of "fishing" depends, however, not only on the bait used but also on the hunger of the various fish, their individual behavior and the information exchanged between them. In other words, the established focus depends not only on the sources of activation (the input and goal lists), but also on the currently active memory elements (the pattern of activation in WM) and the activation exchanged between them. This is due to: (a) the fact that the local extremum found by the relaxation search depends on the starting point of the search, and (b) the fact that in our model we do not wait for a stable state to be established (a

[1] The solution of a given problem starts with the construction of its description. The exact nature of this process is beyond the scope of the present chapter. Lange et al. (1990) discuss how a language understanding process constructs the problem description and Mitchell and Hofstadter (1990) describe the process of problem perception (i.e., the emergence of a problem formulation). Since language understanding and perception are highly context-dependent, these processes contribute to the dynamic aspects of problem solving as well.

local extremum to be found)—but even in a dynamically changing memory state,[6] if the focus is not changed for a definite period of time, it is considered as a retrieved description.

Strategic retrieval in AMBR is used when the reasoner wants to reveal the relation between two concepts or situations at some stage of the reasoning process (i.e., when a search for a path between two nodes in the network has to be conducted for some reason). For example, the mapping process uses strategic retrieval in establishing the semantic similarity between two nodes, whereas the evaluation process uses it in establishing the causal relevance of a memory element (finding a path from that element to a goal node). The former case is described in more detail in Section 5.2. and the latter in Section *5.4.* Strategic retrieval is performed by the marker-passing mechanism.

5.1.2. Explanation of Empirical Facts.

Why is Retrieval (of a source for Analogy) difficult? In order to retrieve a source, the reasoner can partially manipulate her own memory state and rely on the associative mechanism to bring the appropriate description to the focus. As has been explained, the result of the work of the associative mechanism depends on the goal and input lists as well as on all currently active nodes (i.e., the state of the WM). The reasoner can definitely control the nodes on the goal list and at least some of the nodes on the input list, but to control the state of WM is impossible (at least directly). In this way the result of the automatic retrieval is beyond the control of the reasoner. This has both harmful and useful consequences. On one hand, this makes it difficult to retrieve the desired information in an unrelated context, but, on the other hand, in an adequate context this saves efforts for a detailed specification of the retrieval cue and makes the retrieval very effective. The mechanism has some advantages also in unrelated contexts where random creative associations can be reached and in this way unpredictable results obtained.

Why does Semantic Similarity Dominate Retrieval? The associative mechanism used in the retrieval process spreads activation through links, most of which have a definite semantic interpretation. In this way, activation spreads in effect among semantically similar elements and the summation of activation corresponds to the superposition of the overall similarity between cases. Eventually, the focus turns out to be the description most similar to the current situation. The associative mechanism does not distinguish between structural and superficial features so both are used in the retrieval. However, the number of superficial features used to describe a situation is usually greater than the number of structural ones, so superficial features dominate the retrieval. As it

[6] The environment is continuously developing and in this way the input list is also continuously changing, so it is implausible to expect that a stable activation pattern can be established for a definite relatively long period of time. If. however, the changes in the environment are unrelated to the ongoing processes, the focus can stay the same for a longer period of time.

has been shown, the retrieval process depends on the current state of WM (i.e., the elements currently considered as relevant), so an exception may occur if some structural features are considered for some reason highly relevant (being on the goal list or highly preactivated, including being the focus). Then they will have a greater impact on the retrieval process than any superficial features.

Why does the Presence of a Problem scheme make the Retrieval Easier? A problem scheme corresponds to a generalization of several similar problems and in this way it contains generalized versions of the problems' elements and of the relations between them. In this way the semantic similarity between these elements and relations, and the target elements and relations is greater than between the elements of two specific problems. Greater similarity means greater relevance and thus easier retrieval. In the radiation problem,[7] for example, x-rays are more similar to forces then to armies which are a kind of forces.

It must be noted, however, that there may be difficulties in retrieving very abstract schemes because they will be less similar to the target than some more specific cases (e.g., x-rays will be more similar to laser beams than to forces). (The measure of this similarity is obtained by the associative mechanism depending on the particular memory state.)

Priming Effects. There are several possible explanations of the priming effects: (a) the reasoner builds an expectation about the target when presented with the prime and uses this conscious strategy for enhancement of retrieval, (b) the reasoner combines the target and prime at retrieval process, and (c) there is a process of automatic spreading activation.

Posner and Snyder (1975) and De Groot (1983) demonstrate that the priming effects can be caused both by a process of automatic spreading activation and by intentional strategies of the reasoner, like building some general or specific expectations. Moreover, De Groot claims that the former can cause only positive effects while the latter causes both positive and negative effects. Our experimental design does not allow the subjects to build an expectation about the relation between the prime and target as this is the only relation between problems in the whole pool of problem material. On the other hand, the distractor problems (which are unrelated to the target problem) did not cause any negative effect in the control condition (the same results were obtained when the target was the first problem to be solved and when it was preceded by several distractor problems). So we can accept that the priming effects are not of the latter type.

Ratcliff and McKoon (1988) proposed a retrieval theory which assumes that the prime and the target are combined at retrieval into a compound cue that is used to access memory. This theory will, however, have difficulties in explaining the far priming effects demonstrated in our experiments because in these cases the prime does not immediately precede the target and so has to be recalled from memory and identified as relevant.

[7] Used in the experiments of Gick and Holyoak (1983).

So the most plausible explanation of priming effects is that they are due to the residual activation of some knowledge activated during the priming phase.

Long-Term Priming Effect. The priming effects demonstrated in psychological experiments so far are predominantly in low-level tasks and are indeed short-term effects. In my opinion, this is due to the fact that in these experiments only single nodes are activated by the preliminary setting (e.g., only one word) and their activations decay very quickly. In more complex experiments, like the ones reported in this chapter, a large and highly interconnected part of the network is activated and the nodes activate each other mutually for a certain period of time before they calm down (i.e., there is a kind of resonance effect; Rumelhart et al., 1986). That is why the activation of the nodes decreases more slowly and the activation pattern is more stable. Another factor contributing to the long-term effect is that in our model of LTM only excitatory links are allowed.

Decrease of Priming Effect in the Course of Time. Someone may try to explain the demonstrated decrease of the priming effect by interference mechanisms, but the distractor problems are different enough from the prime and the target to be able to interfere with them. Moreover, as the problems and their total number in all conditions are the same (only the order of presentation is different), the total amount of proactive and retroactive interference should be the same. So, in my view it is just an effect of time due to the decay of the activation of the nodes.

A simulation of this kind of priming effect is demonstrated in Section 6 by an implementation of the proposed model.

5.2. The Mapping Process

The objective of the mapping process is: given two descriptions, to find a *correspondence* between their elements, which, on its turn, will determine the transfer of knowledge.

Each time a new focus is established or a new goal becomes the most active one on the goal list, a mapping process is started between this focus and this goal node.

The main issue with mapping is how to overcome the combinatorial explosion which will take place in case of an exhaustive comparison between all possible correspondences. To address this issue, a mechanism for parallel evaluation of possible mappings has to be used as well as a set of constraints restricting the space of possible mappings.

One natural constraint is to put only semantically similar elements into correspondence. This is, however, in some cases too restrictive and in others even not enough:

- sometimes dissimilar elements have to be put in correspondence only because they play similar roles in both situations (i.e., a structural criterion has to be used).

- sometimes there are more than one similar elements in the corresponding descriptions and again a structural constraint has to be used to resolve the ambiguity.

An example of the former case will be a correspondence established between the following two situations: (a) teachers teach pupils about dogs, and (b) pupils teach dogs to respect teachers. Here, "teachers" from the first situation has to be put into correspondence with "pupils" from the second one, while "pupils" from the first situation has to be put into correspondence with "dogs" from the second one.

An example of the latter case will be the analogy between the following two descriptions: (a) on(blockA, blockB) and on(blockB, blockC), and (b) on(cubeL, cubeM) and on(cubeM, cubeN). Here, blockB has to be put into correspondence with cubeM because of the specific binding role played by both of them.

It is also clear that the structural constraint cannot start from nowhere (i.e., a semantic correspondence between two elements has to be already established in order to be able to reflect further the structure within which this element exists or the structure of the element itself).

Finally, the pragmatic importance of the elements of a situation plays a crucial role in establishing a correspondence, thus unimportant elements can be ignored whereas important ones have to be put in correspondence. Moreover, in different contexts different correspondences may be established between the same pair of situations.

Thus, in order to build a flexible mapping mechanism the following three constraints' are used in the present model:

- *structural constraint*—the established correspondence should tend to be an isomorphism (a one-to-one correspondence preserving relations between elements, including attributes),
- *semantic constraint*—the mechanism should tend to establish correspondences between semantically similar elements, and
- *pragmatic constraint*—the mechanism should tend to find correspondences to all pragmatically important (associatively relevant) elements.

A number of mechanisms contribute to the implementation of this set of criteria in AMBR:

1. A node constructor process builds temporary *correspondence* nodes which represent the hypotheses about the correspondence between pairs of elements of the descriptions mapped.

' This set of criteria was first used in ACME (Holyoak & Thagard, 1989b; Holyoak. Melz, & Novick, this volume).

2. The structural constraints are imposed by a mechanism for constructing temporary excitatory and inhibitory links between these corresponding nodes.
3. The pragmatic importance of the elements is measured by their associative relevance as computed by the associative mechanism.
4. The semantic similarity between two elements is measured by the associative relevance of their common super- or metaclass as found by the marker-passing mechanism.
5. The parallel evaluation of possible mappings with regard to all three constraints is accomplished by the relaxation search performed both over the nodes and links constructed temporarily and over the LTM by the associative mechanism.

These mechanisms are discussed in more detail in the following subsection.

5.2.1. Mechanisms of Mapping.

Representation of Mapping and the Node Constructor Process. Each correspondence between two elements, X and Y, is described by a frame representing a proposition of the type "correspond(X,Y)." This frame is placed in a temporary' *correspondence node* which is produced by the node constructor process. The latter builds c-core/links from the correspondence node to X and Y as well as excitatory temporary links *(t-links)* from X and Y to it. There can be excitatory (positive) or inhibitory (negative) *t-links* between the *correspondence* nodes.

Several node constructor processes can run in parallel triggered by various other symbolic processes, such as the processes of establishing semantic or structural correspondence. Because of the limited number of symbolic processors of this type (node constructors) only the most active requests will be satisfied. In this way various processes compete in their attempts to extend the network.

When the mapping process is started, a node called *map* is being constructed and established as a goal. Positive links are built between the map node and all *correspondence* nodes. In this way all *correspondence* nodes are pumped with activity by the goal node. The map node together with all *correspondence* nodes linked to it form the representation of the temporary map. The map that is eventually constructed consists of the most active hypotheses. In the process of learning (see Section 5.5) a permanent version of the map may be stored in LTM.

Process of Establishing Semantic Correspondences. The objective of this process is to evaluate the semantic similarity between two given descriptions or to find semantically similar elements of these descriptions and to put them in correspondence. There are three different cases in the establishment of semantic

' This means it is noi part of the rcasoner's LTM.

correspondences. The most trivial one is the reinstantiation case (DeJong, 1989) when *both descriptions* are proven to be specializations (elements) of a class well known by the reasoner beforehand; the second one is *when pairs of semantically similar* elements of both descriptions are found and put in correspondence; and the third one—when a *new mapping is forced* in order to evaluate the semantic similarity between two important elements. All these cases are discussed in more detail in the following text.

Process of evaluating the semantic similarity between two nodes. The semantic similarity between two nodes (corresponding to objects, relations, situations, etc.) is evaluated by searching for a common super- or metaclass at any level. This increases the possibility of discovering more abstract analogies by finding a very far and abstract common superclass of both descriptions. The main links that are traversed are the *instance-of, is-a,* and *c-coref* links. Thus, two entities are semantically similar if they are identical, refer to the same entity in the world, belong to or are specializations of one and the same superclass (including the case where one of them is a specialization of the other), or if a specific combination of these criteria holds.

The marker-passing mechanism is started from the two nodes corresponding to the descriptions as a whole in order to find a common superclass of both. If such a class is found, then a correspondence node is constructed and the *map* node is connected to it (Figure 5.6). Both paths to this common superclass are traversed in order to establish the slot correspondence.

The associative relevance (the activity) of the common super- or metaclass found is used as a measure for the semantic similarity of these nodes. In this way the degree of similarity between entities depends on the context and can be very different in different cases in our model (see Figure 5.6). For example, two entities having a common superclass can have a high degree of similarity in one context while in another, where this superclass is not relevant, the corresponding degree of similarity can be 0. Thus, there is important pragmatic control exercised on the process of evaluating the semantic similarity.

Process of establishing semantic correspondence between the elements of two descriptions. The purpose of this process is to find pairs of semantically similar elements from two descriptions, to evaluate the degree of this similarity and possibly to'trigger node constructor processes (Figure 5.7).

The process of establishing semantic correspondence between the elements of two descriptions is performed by the second version of the marker-passing mechanism (Section 4.7). It is started from all the nodes referred to by *c-coref* links from the slots of both descriptions in parallel. *Correspondence* nodes are created for similar entities.

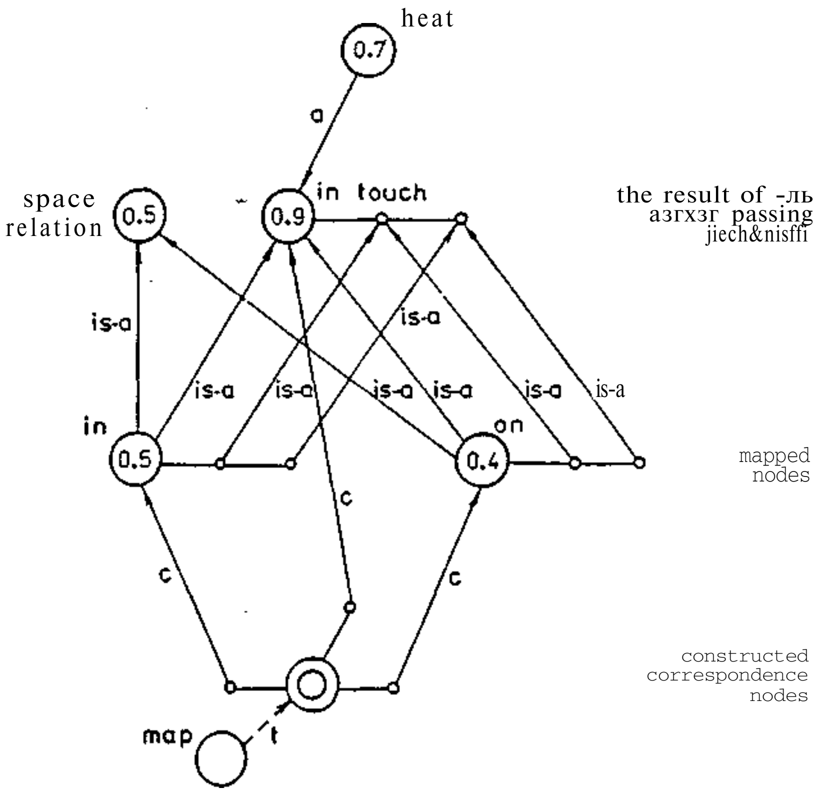
Process of direct mapping between two elements of the descriptions. The aim of this process is to evaluate the semantic similarity between two elements of the mapped descriptions by directly comparing the descriptions pointed to by the corresponding slots (i.e., by triggering an additional mapping process).

Figure 5.6.

Reinstantiation. The marker-passing mechanism finds the most relevant common superclass *(in-touch)* in this particular context and reports its activity. A *correspondence* node is constructed.
Here and in the next figures the following graphical symbols are used:
———— - permanent links,————» temporary links,

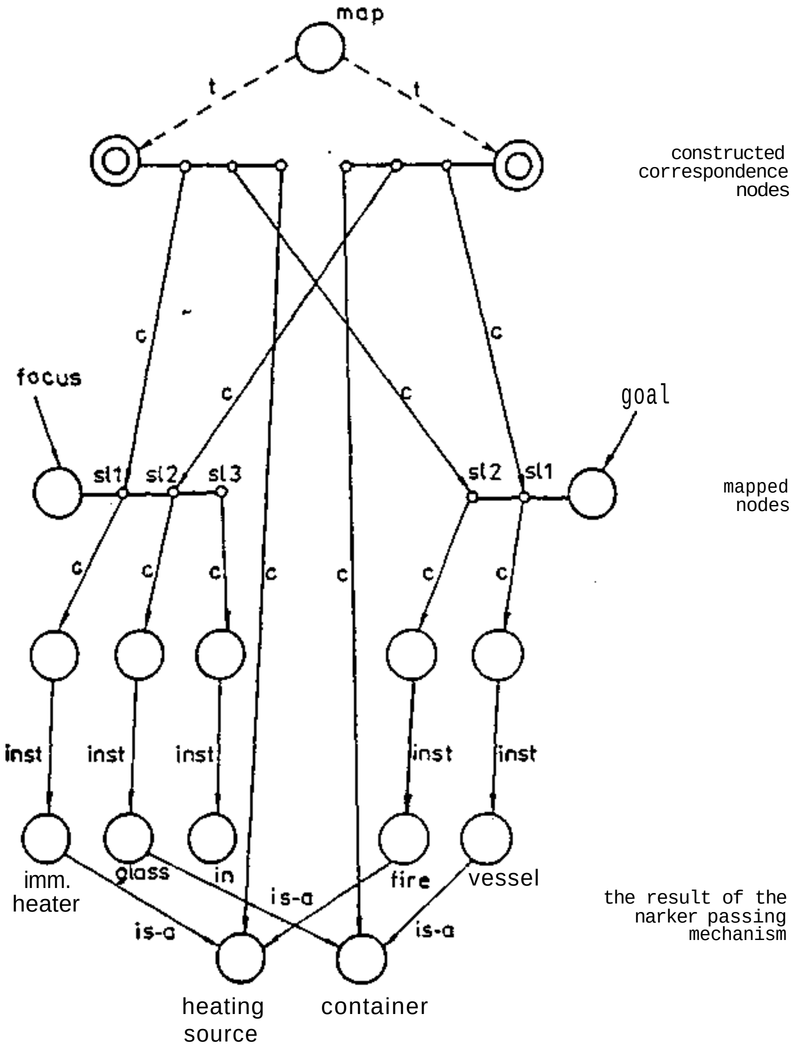
(O) - temporary correspondence nodes, (\ - frame nodes,

O - slot in a frame node



If the most active slot of the target description is not of the *instance-of or is-a* kind, then it is considered to be very important to find a correspondence for this slot. That is why a secondary direct mapping process is started between the filler of that slot and the filler of the most active slot in the source frame. If this mapping succeeds, then a *correspondence* node (within the primary *map)* is built with an activity equal to the activity of the *map* node constructed during the

Figure 5.7.
Establishing Semantic Correspondence between the Elements of Two Partial Descriptions (between the slots of the corresponding frames). The marker-passing mechanism is used to find the common superclasses: "heating source" and "container." Two frames describing the established correspondences are constructed.



secondary mapping process, and an excitatory Jink with a weight of 1 is established from the primary map node to this *correspondence* node. If the secondary mapping process fails, then another secondary mapping can be started with the filter of the next slot of the source (the slots being ordered according to their activity).

Process of establishing structural correspondences. The purpose of this process is: starting with a *correspondence* node, to construct new *correspondence* nodes and links between them on the basis of structural correspondences.

Thus, if a *correspondence* node for two elements in the description is established, the following *correspondence* nodes are constructed:

1. If the two given elements are instances of certain classes, then a *correspondence* node between their classes is constructed, except when they belong to the same class. (This means that if the elements are objects, their classes are put in correspondence, and if the elements are propositions [instances of relations], the corresponding relations are put in correspondence.)
2. If the two given elements are propositions, then *correspondence* nodes are established between their arguments as well (Figure 5.8).

- The two propositions do not need to have the same' number of arguments (or to be ordered in the same way)—during the process of establishing semantic correspondence between the propositions their parts are also put into correspondence (Figure 5.9). This is possible because of the rich representation of relations and propositions.
- If the first relation is a converse relation of the second one, like on(A,B) — support(C,D), then during the process of establishing semantic correspondence the corresponding arguments are found as well.
- If the relations are nonsymmetric ones, like cause(x,y), on(x,y), in(x,y), etc., then *correspondence* nodes are constructed only for their corresponding arguments.
- If the relations are symmetric ones, like in-touch(x.y), married(x,y), then *correspondence* nodes are constructed for each possible argument pairing.

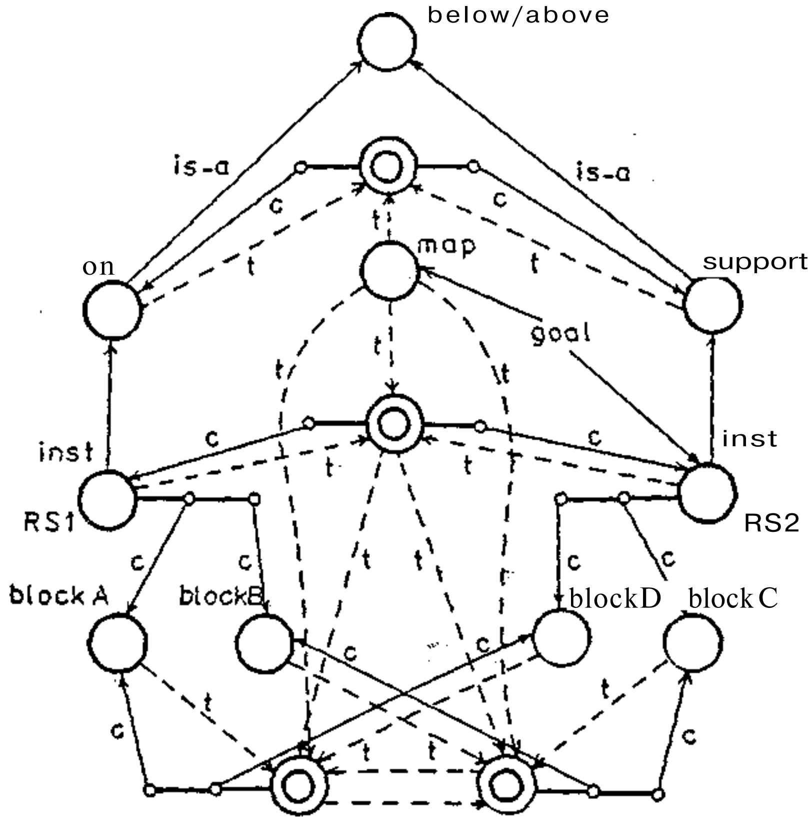
To each created *correspondence* node an initial activation is given which is equal to the activity of the crossroad node (where the marker-passing mechanism has found their correspondence). In this way the activity (the associative relevance) of a *correspondence* node will depend on the associative relevance of the common superclass found rather than on the distance of this class to the elements (the number of links that have to be traversed in order to find it).

Temporary excitatory links are built up between the following nodes:

1. from the map to the *correspondence* nodes,
2. from the elements on the two descriptions to their respective *correspondence* nodes (from x and y to correspond(x,y)),

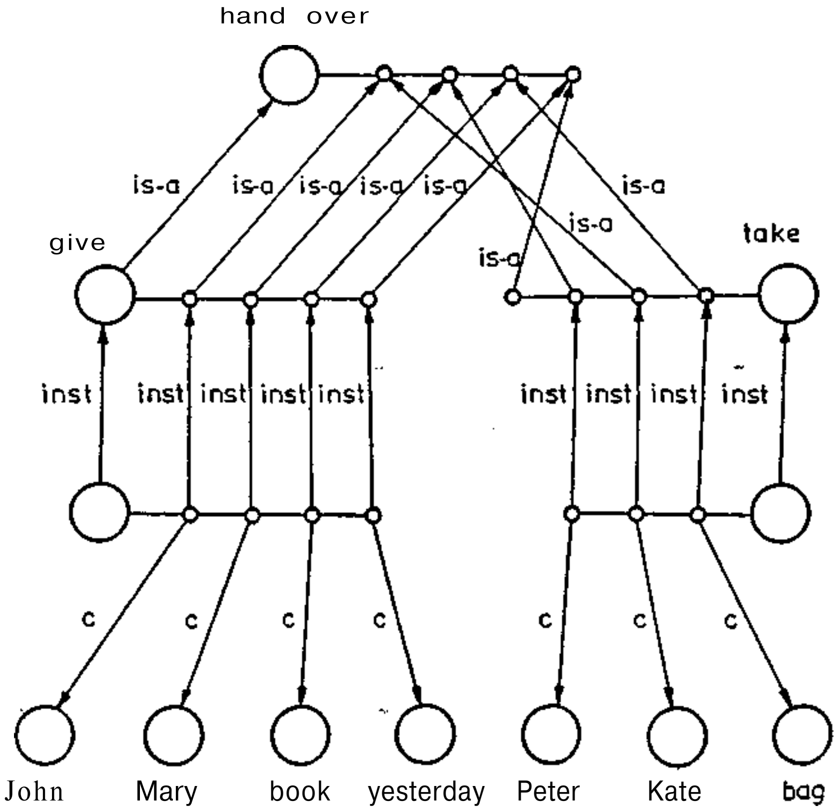
Figure 5.8.

The map built by the structural correspondences process for two relation slots,
RS1: onfblockA.blockB) and RS2: support(blockC.blockD), represents the correspondences established between the propositions RS1 and RS2, the relations on and *support,* and between their arguments blockA and blockD, and blocks and blockC, respectively. Excitatory temporary links are built between the map node and all these correspondence nodes as well as between the correspondence nodes themselves.



3. from proposition *correspondence* nodes to their relation and arguments *correspondence* nodes and vice versa,
4. from one argument *correspondence* node to another, and
5. from relation *correspondence* nodes to argument *correspondence* nodes and vice versa.

Figure 5.9.
Correspondences Established between Relations with Different Number of Arguments:
give(John,Mary,book,yesterday) and take(Peter,Kate,faag)



Temporary inhibitory links are created between competing correspondences.

The weights of all excitatory (respectively inhibitory) links are computed so that:

1. the sum of the weights of all excitatory (respectively inhibitory) links starting from a node is one, and
2. the weights are proportional to the initial activity of the ending nodes.

Parallel Evaluation of the Best Mapping. The best mapping is found by evaluating all possible ones in parallel on a competitive basis. This is done by relaxing the constraint satisfaction network (the LTM extended with the temporarily constructed nodes and links) using the associative mechanism. Because of the links between the elements of the target and source descriptions and the *correspondence* nodes, the activity of all nodes in LTM influences the activity of the *correspondence* nodes. What makes this different from the use of the associative mechanism in the automatic retrieval process is the presence of inhibitory links in the network. After relaxing the network, we find the best mapping in the following way: for each slot in the target the most active *correspondence* node is found (which is the winner on the basis of the competitive mechanism of relaxing the network) and the link between the map node and that correspondence node is strengthened to have a weight of 1, while all other competing links are dropped out. In this way the map node is finally connected only to those *correspondence* nodes which form the best mapping.

The relaxing mechanism does not start at any particular moment, because, as has been pointed out, the associative mechanism runs continuously, and in this way at each moment its partial results are available. That is, at each moment, independently of how much semantic or structural correspondence has been established, the first approximation of a best *map* is present.

Manipulating Input and Goals. One method for pragmatic control of processing is goal manipulation (i.e., putting forward additional goals or subgoals (like direct mapping between most active slots) or rejecting some old goals). Another method for pragmatic control of processing is changing the input nodes. If, for example, something in the environment is changed then the perceptual mechanism will change the state of the input nodes (including the case where another reasoner provides some help by drawing attention to some elements of the target or by presenting additional information). But it is also possible to consider an active reasoner that might, for example, reread a written description of the current problem which would focus her attention on some particular details, or the reasoner might perform actions that change the environment and perceive their effects, or she might write down intermediate results of the reasoning process and then read them back (by redirecting the input activity), or she might manipulate the environment itself by experimenting and testing certain hypotheses.

Pragmatic control on the running ofprocesses within mapping. The mapping process is performed by all those symbolic processes running in parallel. The exact moment when a symbolic processor is triggered, and both its speed and success heavily depend on the activity of the node representing this process and on the activity of the nodes which are used as data by this process. In this way a thorough pragmatic control is established on the way the mapping process is performed.

Processes of establishing semantic and structural correspondence compete with each other running in parallel at different speeds, and thus at each moment both the establishment of semantic correspondences between additional elements and the establishment of structural correspondences by developing the connectionist network further (i.e., building new *correspondences* nodes and links based upon the existing ones) can continue.

Depending on the results of this competition (i.e., which one will be most active) either more concrete or more abstract analogies can be reached. Of course, this competition can be partially influenced by the reasoner by manipulating the input and goal nodes. For example, if a goal to find an abstract mapping is established, the concept "abstract" and the related concepts (like "object," "relation," "cause") will be activated. This will enable the semantic process to find general correspondences (e.g., "John" and "table" both being objects, a property that would not ordinarily be taken into account) and because of the high variability of such correspondences the structural process will dominate the process of finding the best map.

5.2.2. Explanation of Empirical Facts.

The impact of the Degree of Structural Consistency and of Semantic Similarity on Mapping. It is well known that both structural consistency and semantic similarity contribute to the ease of mapping as reviewed in Section 2. So in analogies between remote domains the structural consistency dominates whereas in intradomain analogies (case-based reasoning) the semantic similarity is very useful.

If, for example, the target and the source are from one and the same domain (or from close domains) then they will share a lot of objects of the same class, a lot of relations, etc. This will highly activate these classes and relations and the semantic process will dominate over the structural one. In the case of remote analogies there will be no common classes and relations that are sufficiently activated and that is why the structural process will dominate. So the final result will depend on the competition between the processes of establishing semantic and structural correspondences (i.e., on the context in which they run).

Why is it Difficult to Establish a Mapping between Analogs from Remote Domains? The main reason for this difficulty is the low degree of semantic similarity between the two descriptions, with regard to both the objects and the relations. This causes an enormous growth of the set of possible mappings where the structural constraint has to dominate, but even to be able to apply the latter, a semantic correspondence between some elements (e.g., higher order relations) has to be already established.

Moreover, in order to consider two elements from remote domains as similar (i.e., as members of one and the same class), an unusual (less typical) point of view has to be used (i.e., the activation should pass through weaker links) and therefore in order to accumulate enough activation for that class a greater number of sources linked to it has to be used. So a specific context in which this class is highly relevant will be of great help.

Why does Cross-Mapping Impair the Establishment of Correct Correspondences between Source and Target. Especially When an Overall Similarity *between Them is Present?* In the case of cross-mapping (i.e., when similar objects play different roles in the analogs, there is a conflict between the processes of establishing semantic correspondence and of establishing structural correspondence, so each of them destroys the work of the other. Let us consider the following example:

Source: Mary embraced Robert.
Target: John kissed Kate.

If the semantic correspondence established between embrace and kiss (both are kinds of loving actions) is the most active one then the structural correspondence process will force a correspondence between the agents and the recipients of the actions, correspondingly. If, however, for some reason Mary or John are with higher activation levels than the actions, then the correspondence established between Mary and Kate (both female persons) or between Robert and John (both male persons) will dominate over the structural correspondence. In this way, depending on the context,"the establishment of correct correspondences may be impaired. Moreover, the greater the overall similarity between the analogs, the higher the activation of the semantically similar concepts (because they mutually activate each other through the links in LTM), and therefore the semantic constraint prevails over the structural one and in this way the chances of a successful mapping are lessened.

5.3. The Transfer Process

The purpose of the transfer process is to extend the mapping between two descriptions by constructing new slots in the target description that correspond to unmapped slots in the source and in this way to transfer new knowledge to the currentsituation.

Two kinds of slot transfer in AMBR will be considered in this subsection: transfer of an aspect slot and transfer of a relation slot (action slots will not be considered for the time being). When a relation slot is transferred, it is actually the corresponding proposition that is transferred (e.g., P(x) or />(xl, .., *xri),* where *P* is a relation described by another description frame and *χ, χ], ..,χη* are aspect slots with *c-coref* links to the descriptions of the fillers). For simplicity, only propositions of type *P(x)* will be considered further; everything said about them will concern relations of arbitrary arity as well. The following cases can be considered:

1· If, during mapping, both the aspect slot .r and the relation *P* are mapped on x' and P' respectively, then the slot corresponding to the proposition *P(x)* is easily transferred to a new slot with a *c-coref* link to a description of the proposition $P'(x')$, that is—

$x \rightarrow x'$
$P \rightarrow P'$

$P(x) \rightarrow P'(x')$

If, however, during mapping no correspondence is found forд: or *P,* then the reasoner if faced with more difficult problems as discussed in Cases 2 and 3.

2. If the object correspondence is known but the relation correspondence is unknown, that is—

PM -» ?

then either the same relation can be transferred: *P(x)* -* *P(x'),* or a reinstantiation *P'* of some superclass *Q of P* can be produced: *P(x)* —*· *P'(x'),* or the generalization β can be used: *P(x)* -> Of*')· The particular decision would depend on the associative relevance *of P, Q,* and *P',* and on the decisions of the evaluation process. The above is illustrated by the following examples:

water —> coffee
container -* cup
in -* *?*

in(water, container) —> in(coffee, cup)

and
container -* stone ,*.,.
plate —> fire
on —* *?*

onfcontainer, plate) —* in(stone, fire)

In the first example the marker-passing mechanism starts the search from *coffee* and *cup* traversing *c-coref* links in order to find in which relations they can both participate and *in* turns out to be the most typical one (thus, receiving greater activation).

In the second example the marker-passing mechanism starts the search from *fire* and *stone* and the relation *in* is found both because of its direct c-core/links to fire and stone, and because it is a reinstantiation *of on* and is therefore highly activated. Possible common superclasses of *in* and *on* are *space relation* and *in touch:* the second one is considered as more relevant in the presence of objects *tike fire and plate* related to heating.

3. If the relation correspondence is known, but the object correspondence is unknown, that is—

$x \rightarrow ?$
$P \rightarrow P'$

P(x)-> ?

then both a new aspect slot *y* and a new relation slot are constructed. The filler of the relation slot will be *P'(y)* and a filler of the aspect slot will be sought (possibly after the transfer of some other propositions referring to that slot) which: (a) will satisfy the argument restrictions of all transferred relations, and (b) can be found in the target situation (the evaluation process decides whether these two conditions are met). In particular, a reinstantiation of д: can be sought (i.e., another instance of a class to which *x* belongs).

As an illustration, let us consider the following problem: an analogy has to be drawn between the Situation A, in which a lamp on a table is illuminating it, and the Situation B, where a lamp is not available (Figure 5.10).

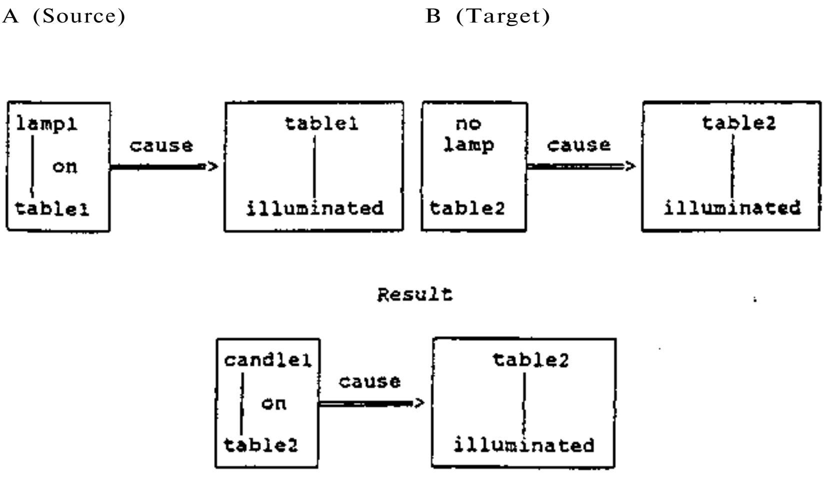


Figure 5.10.
Transferring a Relation with an Unknown Argument. Here the relation *on* is transferred and a substitute for lamp is sought.

In this case we will have:

table1 -> table2
on —> on

on(x,y) → on(x',y')
c-coref(y,table1) → *c-coref*(y',table2)
c-coref(x,lamp1) → *c-coref*(x',candle1),

because both *lamp* and *candle* are subclasses of the *light source* class which has been activated by the presence of light in the situation.

The selection problem in transfer (the problem of which unmapped slots to transfer and which to leave out) is solved on the basis of the pragmatic importance of elements of the source description (both their associative and causal relevance are used). As it has been demonstrated above, however, some structural constraints can force additional transfer (e.g., thejransfer of a relation causes the transfer of its arguments).

Transfer is essentially performed by the same (or similar) processes as mapping, so the construction of new propositions is done by a structural correspondence process and a node constructor. The search for candidate correspondences is done by the associative mechanism, possibly forcing a new mapping process. Reinstantiation is performed when the input or the associative mechanism provide a candidate which is then used by the marker-passing mechanism in order to establish semantic correspondence.

Usually transfer is done actively using the results of the evaluation process.

5.4. The Evaluation Process

The objectives of the evaluation process are to evaluate the maps (including the partial ones) produced by the mapping process and to estimate the consistency, validity, plausibility, relevance, and applicability of inferences made by the transfer process.

Maps can be evaluated on the basis of local as well as global criteria. Local map evaluation is aimed at finding inconsistent correspondences established during mapping (e.g., *red(x)* - *white(x)* (on the ground that both are colors), *hot(x)* - *cold(x)* (both are temperature states). An analysis of causal relevance of the corresponding propositions is needed, because color may be irrelevant while temperature may be relevant, and in this case the first pair will be consistent while the second will not. The global evaluation of the map may be obtained on the basis of a measure of the constraint satisfaction level achieved by the network of correspondence nodes—this measure is given by a gain function *G*, where *G*

$= \Sigma_i \Sigma_j$ Wij $a_i(t)$ $a_j(t))$[10]. If inconsistent but causally relevant correspondences are found or the global evaluation of the map is below a certain threshold, then the map is rejected.

Consistency of inferences means coherence and lack of contradictions within the extended description of a situation. The evaluation process has to check if the filler restrictions are fulfilled when an object is sought for proposition construction in the transfer process and whether the constraints on the interdependences between slots hold.

The validity of the inference itself has to be checked, that is, whether contradictions exist between the inference proposed and the information known about the world (the reasoner's domain knowledge). This may force a new mapping which will establish (or possibly fail to establish) a correspondence between this inference and an element of another description in the target domain. Let us consider the following example. A solar eclipse occurs when the moon is between the sun and the Earth. By an analogy an inference can be made that a lunar eclipse occurs when the sun is between the moon and the Earth, which is quite consistent by itself but invalid compared to other astronomical knowledge.

Plausibility is the reasoner's estimation of the possibility of an inference being true (when no explicit domain knowledge is present and the validity cannot be checked), that is, it is the certainty factor of an inference. The reasoner's certainty in the inferences heavily depends on her estimation of the established mapping (global map evaluation). Some experiments performed by the author (Kokinov, 1992) show a definite dependence between the goodness of mapping correspondence and the certainty of inference. Moreover, by manipulating different elements of this correspondence—objects, components, relations, and properties—we receive different results.

An analysis of causal relevance can be used in order to find out which inferences to keep and which to drop out. This is performed by the marker-passing mechanism starting both from the inference and the goal and traversing only the *cause* relations (i.e., at every node it searches for a relation slot pointing to (by a c-cwe/link) an instance of cause relation). If an element is found to be causally relevant, it is additionally activated as well.

Applicability of transferred actions (plan steps) means checking preconditions of actions and exploring restrictions and resource limits. This is performed by starting additional mappings.

[10] It will be useful to think of the relaxation search in the constraint network as a maximization of the function C, although there is no formal proof of that. This formula is similar to that used by Rumelhart et al. (1986).

5.5. The Learning Process

Learning occurs at different places: (a) during problem solving (it keeps traces of the problem-solving activity and its intermediate results); and (b) after the problem is solved it stores the results and possibly makes some generalizations.

The maps produced by the mapping process can be stored for future use, so that if the same descriptions were compared, the mapping process would not be needed. The permanent map will approximate the settled constraint network. Only the most active *correspondence* nodes (the winners) are permanently stored in LTM together with the excitatory links between them. Inactive (or less active) nodes as well as inhibitory links are not learned.

Failures during the problem-solving activity can also be learned in order to avoid them in the future. This includes produced but rejected maps, produced but rejected inferences, retrieved but unsuccessfully mapped sources, evaluations of transfer, etc.

A problem solved successfully can be a good source for future analogies. That is why the target description together with all inference nodes are stored in LTM. All the links connected to this description (and used during the reasoning process) are strengthened. This makes it possible to retrieve the latter in a future situation. Also, the links between the *map* node and the correspondence nodes are strengthened; thus, if the map is frequently used, these links will acquire the maximum strength of 1, the *map* node becoming a description frame rather than being loosely connected with the *correspondence* nodes. Nodes which are frequently used will lessen their activation thresholds, which will make them more sensitive and enable more processes to access them.

Finally, a generalization of the problems and their solutions can be easily made using the map and the results of the marker-passing mechanism after it has found all common superclasses of the corresponding elements of both descriptions. Starting with these common superclasses and one of the two descriptions, a mapping and transfer process can construct a generalized description. (Note that this generalization is performed exactly by the same reasoning process that supports analogy.)

5.6. Pragmatic Control on the Running of Processes

AH processes described above are activated independently and can run in parallel with others. Moreover, it is also possible that several copies of one and the same process run in parallel. Control over the running of those multiple processes is exercised to pragmatic considerations.

Mapping is triggered by the change of the focus (when the new focus has stayed the same for a sufficiently long period of time). In this case a new mapping process is started between the new focus and the goal (the target description). If such a mapping between exactly the same structures already exists, then it is continued instead of starting a new mapping.

A transfer process can be started after the best mapping is found, but in some cases (if it is highly relevant) it can be started prior to the completion of the mapping, using only partial results from the ongoing process of mapping.

Evaluation can be started after the transfer of new knowledge is finished, but it is also possible to evaluate each single transfer (before the transfer process proceeds). As Keane (1988) points out, usually in a familiar target domain the evaluation is done during transfer, while in unfamiliar domains often a hypothesis needs to be tested in the real world after the transfer is complete.

Learning runs in parallel to all other processes, enabling the storing of maps, analogical inferences, evaluations or failures, and all other kinds of intermediate results. It can, however, start after the problem is solved as well, in order to store the target description together with the solution, possibly to make some generalizations and store them, and to adjust link weights in order to provide for better retrieval in the future.

All triggering of these symbolic processes as well as their termination and speed are controlled by the associative mechanism (i.e., by pragmatic factors).

6. A COMPUTER SIMULATION

A computer implementation of AMBR has been developed that simulates human problem solving in the area of cooking and boiling water, eggs, etc., in the kitchen or in the forest. The simulation system demonstrates AMBR's capability of analogical problem solving as well as some of the priming effects found in the experiments described in Section 2.

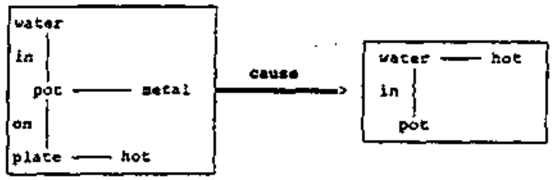
The Simulation Program has been developed in Common Lisp on an AT/286-type computer with 6MB of RAM.

6.1. The Knowledge Base of the Simulation and the Target Problem

The knowledge base of the simulation program contains about 300 nodes and 4,000 links. There are about ten situations related to water, three of which are the following: (Situation A) heating water on the plate of a cooking stove in a pot (Figure 5.1 la), (Situation B) on the fire in a wooden vessel (Figure 5.lib), and (Situation C) heating water by means of an immersion heater in a glass (Figure 5. lie).

A simplified formulation of the target problem given in the psychological experiments is used as a test example in the simulation: *how can you heat water in a wooden vessel when you are in a forest, having only a knife, a match-box and on axe.* The problem is represented in the following way: the reasoner should

A

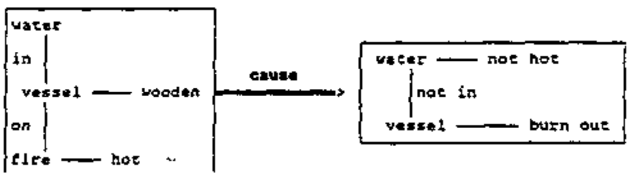



```
G1 <heating water on a plate>
   slot1:  c-coref: G101 <instance of water>
   slot2:  c-coref: G102 <instance of pot>
   slot3:  c-coref: G103 <instance of plate>
   slot4:  c-coref: G104 <in slot1 slot2>
   slot5:  c-coref: G105 <on slot2 slot3>
   slot6:  c-coref: G106 <metal slot2>
   slot7:  c-coref: G107 <hot slot3>
   slot8:  c-coref: G108 <and slot4 slot5 slot6 slot7>
   slot9:  c-coref: G109 <hot slot1>
   slot10: c-coref: G110 <and slot4 slot9>
   slot11: c-coref: G111 <cause slot8 slot10>
```

```
G104 <instance of relation in>
   slot1:  c-coref: G1.slot1
   slot2:  c-coref: G1.slot2
```

```
G105 <instance of relation on>
   slot1:  c-coref: G1.slot2
   slot2:  c-coref: G1.slot3
```

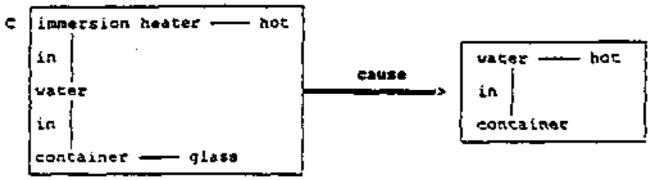
B




```
G2 <heating water on a fire>
   slot1:  c-coref: G201 <instance of water>
   slot2:  c-coref: G202 <instance of vessel>
   slot3:  c-coref: G203 <instance of fire>
   slot4:  c-coref: G204 <in slot1 slot2>
   slot5:  c-coref: G205 <on slot2 slot3>
   slot6:  c-coref: G206 <wooden slot2>
   slot7:  c-coref: G207 <hot slot3>
   slot8:  c-coref: G208 <and slot4 slot5 slot6 slot7>
   slot9:  c-coref: G209 <not hot slot1>
   slot10: c-coref: G210 <burn_out slot2>
   slot11: c-coref: G211 <not in slot1 slot2>
   slot12: c-coref: G212 <and slot9 slot10 slot11>
   slot13: c-coref: G213 <cause slot8 slot12>
```

```
G204 <instance of relation in>
   slot1:  c-coref: G2.slot1
   slot2:  c-coref: G2.slot2
```

```
G205 <instance of relation on>
   slot1:  c-coref: G2.slot2
   slot2:  c-coref: G2.slot3
```




```
G3 <heating water by immersion heater>
   slot1:  c-coref: G301 <instance of water>
   slot2:  c-coref: G302 <instance of container>
   slot3:  c-coref: G303 <instance of immersion heater>
   slot4:  c-coref: G304 <in slot1 slot2>
   slot5:  c-coref: G305 <in slot3 slot1>
   slot6:  c-coref: G306 <glass slot2>
   slot7:  c-coref: G307 <hot slot3>
   slot8:  c-coref: G308 <and slot4 slot5 slot6 slot7>
   slot9:  c-coref: G309 <hot slot1>
   slot10: c-coref: G310 <and slot4 slot9>
   slot11: c-coref: G311 <cause slot8 slot10>
```

```
G304 <instance of relation in>
   slot1:  c-coref: G3.slot1
   slot2:  c-coref: G3.slot2
```

```
G305 <instance of relation in>
   slot1:  c-coref: G3.slot3
   slot2:  c-coref: G3.slot1
```

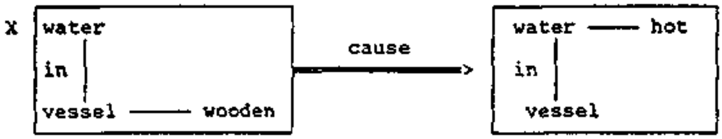
Figure 5.11.

Part of the Situation Descriptions in the KB of the Simulation:
a) Situation A. Heating water on the plate.
b) Situation B. Heating water on the fire.
c) Situation C. Heating water by an immersion heater.
A graphical representation as well as a simplified frame representation of each situation are given.

look for a situation in which the water is in a wooden vessel and which will cause another situation in which the water will be hot and will still be in the wooden vessel (Figure 5.12).

The links between the nodes corresponding to these situations (e.g., Gl) and the nodes corresponding to all concepts referred to in these descriptions (G101, GI02, etc.) are weighted (the weights are not shown on the figure for simplicity). There are also weighted *a-tinks* in the reverse direction (e.g., from water to G101, and from G101 to Gl). The weight of a link depends on the typicalness of the corresponding relation (i.e., how often it is used). For example, the weights of the links between water and heat on the one hand and Situation A on the other hand are greater than that connecting water and heat w'th Situations B and C, because A is a more typical situation then B and C.

Figure 5.12.
Representation of the Target Situation (put on the goal, list to start the problem-solving process).




```
G4 <targetproblem: heating water in the forest>
   slotl:  c-coref: G401 <instance of Kater>
   slot2:  c-coref: G402 <instance of vessel>
   slot3:  c-coref: G403 <in slotl slot2>
   slot*:  c-coref: G404 <vrooden slot2>
   slots:  c-coref: G405 <and slot3 slot4>
   slot6:  c-coref: G4Q6 <hot slotl>
   slot?:  c-coref: G407 <and slot3 slot6>
   slots:  c-coref: G408 <cause slots slot7>

G403 <instance of relation in>
   slotl:  c-coref: G4.slotl
   slot2:  c-coref: G4.slot2
```

6.2. Simulation Experiments

The target problem described above has been presented several times to the system. The representation of the problem and the input have been (almost) the same each time; the conditions have differed in the system's preliminary setting.

In the first experiment only the target problem is given to the system without any preliminary setting. Its description is connected to the goal node and several descriptions are connected to the input node, namely the descriptions for heat, water, vessel, wooden, knife, matchbox, axe, forest (Figure 5.13-la).

Starting from this state the associative mechanism will bring the system to a new memory state where a new description, G1, will become the focus. It is the description of the most typical situation, A, connected to *heat* and *water*—the sources with greatest activation (capacity). The description G2 corresponding to Situation B is also highly activated but less than Gl (the links from matchbox to *fire,* from *fire* to G203, and from G203 to G2 as well as from *water, vessel, wooden, heat* to G2 are used).

At this place a mapping process starts between the goal node (G4) and the new focus (Gl). The correspondences found by the semantic similarity criterion (the marker-passing mechanism) are:

Gl.slotl <water>—G·*.slotl <water>
Gl.slotZ <pot>—G4.slot2 <vessel> (both are containers)
Gl.sloto <metal>—G4.sloi4 <wooden> (both are materials)

Figure 5.13.

The state of the Memory of the Simulation System in the Various Experiments. (G1, G2, G3, G4—the same as in Figures 5.11-5.12)

E x P No	Sources of activation		Working Memory	
	input nodes	goal nodes	current focus and other active nodes	new focus and other active nodes
la	(water *O.S)* (vessel 0.6) (wooden 0.7) (heat 0.9) (forest 0.4) (knife 0.5) (axe 0.2) (match-box .4)	(G4 0.9)	arbitrary, unrelated to the target problem	(Gl 0.8) (fire *0.6)* (plate 0.7) (G2 0.6) (G3 0.1)
Ib	(water 0.8) (vessel 0.6) (wooden 0.7) (heat 0.9) (forest 0.4) (knife *0.5)* (axe 0.2) (match-box .4)	(G4 0.9)	(fire 0.8)	(G2 0.8) (fire 0.7) (Gl 0.6) (G 0.1)
2a	(water *0.8)* (vessel 0.6) (wooden 0.7) (heat 0.9) (forest 0.4) (knife 0.5) (axe 0.2) (match-box .4)	(G4 0.9)	(imnersion heater 0.9) (water 0.3) (hot 0.8)	(G3 0.9) (Jcnife 0.6) (imm.h. 0.8) (water 0.7) (hot 0.7) (Gl 0.8)
2b	(water 0.8) (vessel 0.6) (wooden 0.7) (heat 0.9) (forest 0.4) (knife 0.5) (axe 0.2) (match-box .4)	(G4 0.9)	(immersion heater 0.7) (water 0.6) (hot 0.6)	(G3 0.81) (knife 0.6) (imm.h. 0.7) (water 0.7) (hot 0.7) (Gl 0.8)
3	(water 0.8) (vessel 0.6) (wooden 0.7) (heat 0.9) (forest 0.4) (knife 0.5) (axe 0.2) (match-box .4) (stone 0.7)	(G4 0.9)	(imnersion heater 0.9) (water 0.8) (hot 0.8)	(G3 0.9) (knife 0.6) (stone 0.8) (imm.h. 0.8) (water 0.7) (hot 0.7) (Gl 0.3)

{Gl.slor9, Gl.slot?} <insiances of *hot*>—G4.slo(6 <instance of *hoi*>

{Gl.slot4, Gl.slotf}—G4.slot3 (instances of the *in* relation or more generally of the *in-touch* relation)

{Gl.slotS, Gl.slotlO}—{G4.slot5, G4.slot7} (instances *of and)*

G1.slotll <cause>—G4.slot8 <cause>.

(The lists in curiy braces ({...}) represeni ambiguities in the found correspondences that cannot be resolved using only [he criierion of semantic similarity.)

i

Correspondence nodes are constructed for these pairs as well as for their arguments or metaclasses (now reflecting the structural constraints) by the node constructor as soon as a semantic similarity is established. The initial activation of these nodes as well as the weights of the temporary (excitatory and inhibitory) links built between them reflect the activation level of the crossroad node found by the marker-passing mechanism.

The associative mechanism (continuously running in parallel) now functions as a constraint satisfaction machine working over the LTM extended with the newly created nodes and links, resolves the ambiguities and finds the best map (the most active correspondence nodes):

Gl.slotl—G4.slotl
Gl.slot2—G4.slot2
Gl.sloto—G4.s]ot4
Gl.slot9—G4.slot6
Gl.slot4—G4.slot3
G1.slotS—G4.slot5
Gt.slottO—G4.slot7
Gl.siotll—G4.slot8

At this place the transfer process starts to seek a correspondence for the highly activated slotS of Gl <a plate>. The concept *plate* and the relations *on* and *hot* are put on the goal list. The associative mechanism brings G203 <*afire*> in the focus, participating in all these roles. Now the system is getting in the state depicted on Figure 5.13-lb. Starting from this state the associative mechanism brings G2 <Situation B> in the focus and a second parallel mapping is started between G4 and G2 (the current transfer process continues to build a new slot of G4—slot9: G409 <a fire>—corresponding to slot3 of Gl).

in short, the result of this second mapping is the following correspondence:

G2.slotl—G4.slotl
G2.slot2—G4.slot2
G2.slot4—G4.slot3
G2.slot6—G4.slot4
G2.s!ot8—G4.slot5
G2.slot9—G4.slot6
G2.slotl2—G4.slot7
G2.slotl3—G4.slots
G2.slot3—G4.slot9

the transfer process builds correspondences to slot5, slot?, slot10, and slotl 1 of G2 and in this way G4 becomes an instantiation of G2 (i.e., the reasoning process has switched to a deductive one).

The evaluation process detects an inconsistency between G2.slot9 <not hot water> and G4.slot6 <hot water> and reports that in this way the water cannot be heated (i.e., in the end the system fails to solve ' 'arget problem).

In the second experiment, a set of concepts (immersion—heater, water, hot, etc.) are presented to the system for a definite period of time before the target problem is given. As a result the system starts the reasoning process in the state described in Figure 5.13-2a.

Starting from this memory state, the associative mechanism brings the description G3 (i.e., Situation C, in the focus). A mapping between G3 and 04 starts which results in the following correspondences:

03.slot1—G4.s/ocl
G3.slol2—C4.slol2
G3.slot4—G4.slot3
G3.slot6—G4.s!ot4
G3.slotS—G4.slot5
G3.slot9—G4.slot6
G3.slotlO—G4.slot7
G3.slot11—G4.slotS

At this point the transfer process starts to seek a correspondence to the immersion heater. It puts *immersion heater, hot,* and *in* on the goal list and waits for the result of the associative mechanism. Actually, there is nothing very close to these constraints in the memory of the system, so the result of this retrieval will depend more on the overall state of the memory than on this specific cue. Since the concept of a knife is on the input list (with relatively high activity) and it receives additional activity from *immersion heater* (via the *object* node) and from *in* (because a knife, being small, can be put in other objects), it eventually becomes the focus. The evaluation process starts a marker passing for knife and hot and detects a path through a heating plan (the knife itseif has to be heated in the fire). So. now the transfer process constructs an additional slot—slot9: <knife>—in G4, and transfers the slots for *hot* (slot?) and for the relation *in* (slotS) from G3 to G4. In this way, a complete correspondence between G3 and G4 is established and the target problem is solved.

in a second run of the program the preliminary setting of the system is finished two minutes" before the target problem is given (i.e., the concepts for immersion heater, water, and hot are disconnected from the input list two minutes before the input for the target problem is connected). This results in decreasing their activity until the problem solving starts, but they are still active enough so that, as depicted in Figure 5.13-2b, the system still gets the G3 description in the focus when presented with the target problem. In this case the system replicates more or less the above described behavior and finds the same solution of the problem. If, however, the time delay for presenting the target is

" The particular real-time intervals in the psychological and simulation experiments should not be compared literally because the computer simulation, being a specialized system, solves the problem much more 4ly than a human reasoner.

greater, than G1 again becomes the focus and the system fails to find a solution of the problem.

In the third experiment, the starting state of the system is the same as in Experiment 2 differing only in the input: the concept of a stone is on the input list as well (Figure 5.13-3). This corresponds to a situation where a stone is perceived (or imagined) by the reasoner during the problem-solving process.

Although unrelated to the problem description, it is activated by the input and plays a role in the reasoning process. Thus, for example, when the retrieval process searches for a substitute for the immersion heater the node for stone turns out to be more active then the node for knife or axe (receiving activation both from the input and from the *forest* node) and becomes a focus. In this case the system succeeds in finding a solution again, proposing to use a hot stone for heating the water.

6.3. Comparing the Simulation and Experimental Results

6.3.1. Difficulties in Analogical Problem Solving. Analyzing the problem-solving process in this simulation, we find two critical points: the retrieval of the Situation C as a base for analogy and the retrieval of the concept of a knife (or a stone) as a possible correspondence to the concept of an immersion heater. Both critical points concern the retrieval mechanism, because what it produces depends not only on the input and the goal (the target problem), but on the current state of WM (i.e., on the internal state of the system at the given moment) as well. These critical points correspond to the difficulties observed in human problem solving as demonstrated by the psychological experiments described in Section 2.1.

6.3.2. Priming Effects. It has been demonstrated that different results can be obtained by the simulation, even starting with the same inputs and goals, by different preliminary settings of the system. In the first simulation experiment, it has been shown that the system (similarly to human reasoners) has difficulties in retrieving the proper source when presented with the target problem without any preliminary setting. In the second experiment a replication of the priming effect described in Section 2.2. is demonstrated: a preliminary setting with the "immersion heater situation" helps the system to find a useful source and successfully to solve the target problem. The far priming effect as well as the decrease of the priming effect in the course of time are replicated as well.

In the third experiment the role of the environment is explored. When a stone is presented to the input of the system during the reasoning process a different solution of the problem has been found (using a hot stone). This can be considered as a prediction made by the model about the behavior of human reasoners in similar situations.

These experiments reveal the variable context-dependent behavior of the simulation system.

7. COMPARISON TO RELATED WORK

The current work has a lot of precursors and sources of ideas. I will try to mention at least some of them. Related architectures will be reviewed initially, and then some related models of analogical reasoning.

7.1. Hybrid Architectures

In recent years interest in hybrid architectures has emerged and grown (Barnden & Pollack, 1991; Hendler, 1989; Stark, 1991). Several examples taken from contemporary research are considered later.

One of the well-known proponents of this approach is Hendler (1989a, 1991). He regards marker-passing and connectionism as two different types of spreading activation (spreading symbols and numbers, respectively), proposing a hybrid planning system which combines both of them. Actually, his system consists of two *separate modules* which interact with each other. The first module is a semantic network with a marker-passing mechanism running over it and the second one is a connectionist network with numeric activation spreading over it. These two networks have common nodes which are used for interaction between the modules: when such a node is being marked, it then becomes a source of activation for the connectionist network, and vice versa—when activated, it starts to spread a symbolic marker. In this way only one type of marker can be used in the semantic network.

In contrast to the above mentioned architecture and according to my notion of dualism and consistent hybrid approaches (see Section 1.2), marker passing and spreading of numeric activation are considered just as two dimensions of a single network (the reasoner's LTM) in the architecture presented in this chapter. In this way a single node has always an associated activation level and can hold a number of different markers. This makes it possible to establish simultaneously the semantic correspondences needed by the mapping process and even to run several marker-passing processes in parallel.

In order to limit the number of marked nodes a kind of attenuation mechanism is used in Hendler's architecture. It involves an "energy" numeric value (called zorch) associated with each symbolic marker, which is then divided by each node's outbranching as marking proceeds. In this way both the length and the outbranch of the paths found are limited. In my view, this approach has two shortcomings. First, it seems redundant to use two different numeric activations (zorch and connectionist activation), even in a hybrid architecture, when a single one would do the job. Second, this criterion for path preference is too rigid and not well grounded psychologically. In contrast, a context-dependent path preference criterion is used in our architecture using a single activation value (so it is also possible for a long and highly outbranching path to be preferred in a very specific context).

Another hybrid approach involving both marker passing and spreading activation has been proposed by Lange and Dyer (1989), Lange, Melz, Wharton, and Holyoak (1990), Dyer (1991). They, too, like Hendler, use two different networks—one for markers (the so-called top plane) and another for spreading activation (the bottom plane), but they consider both networks as parallel (i.e., with definite correspondences between their nodes and links). The nodes in the bottom plane represent frames (concepts) and the links represent the semantic relations between them. The nodes on the top plane correspond to the concepts' markers (identifiers) or to binding nodes for the frames' roles, and the links connect binding nodes that have to be bound by the same markers. A marked binding node represents a role binding, whereas the activation of a concept node represents the amount of evidence available for that concept. Evidential activation spreads through the semantic links (i.e., through the network in the bottom plane), inferences are made by propagating the markers through the top plane. A special feature of this architecture is that all possible inferences are being derived in parallel (i.e., several different markers are being bound to a role simultaneously), while it is the evidential activation of the corresponding conceptual nodes which determines which one to be preferred (which interpretation to be chosen) in the particular context. This feature is best used for disambiguation and reinterpretation in a natural language understanding process. This is a promising approach in dealing with context and it would be interesting to apply it to the reasoning process as well.

It seems, however, a little wasteful to compute *all* possible inferences with no regard to their relevance to the current context. In the architecture presented in this chapter it is also possible to perform several mappings in parallel, but these are definitely not all possible mappings, but only those that received some support from the particular context. Moreover, they run at different speeds and can use different amounts of resources (available data and node constructor processes) depending on their relevance factors. In addition, the associative mechanism continuously restructures the network reflecting the current context thus changing the set of possible inferences.

An important advantage of the architecture proposed by Lange and Dyer (1989) is that, although hybrid, it is presented in a consistent connectionist way, where markers are represented by signatures—activation patterns uniquely identifying a concept.

There are works which are usually referred to as symbolic but which can be considered as precursors of contemporary hybrid systems. These are symbolic architectures which use spreading graded activation.

One well-known example of such a cognitive architecture is ACT* (Anderson, 1983). This work has strongly influenced my research on cognitive modeling. ACT* involves a hybrid representation scheme: a semantic network for the declarative knowledge and rules for the procedural knowledge. Recently, a new version, PUPS (Anderson & Thompson, 1989), has been proposed which uses schema-like representations for the declarative part. In both architectures a spreading activation mechanism is used over this part in order to select the relevant knowledge. Thus, all production conditions are matched only against active facts (in ACT*) and schema-like structures (in PUPS). Moreover, conflict resolution is determined by activation-based pattern matching (i.e., the more active matches are preferred).

Another architecture of a similar type is PI (Holland et al., 1986; Holyoak & Thagard, 1989a). Schema-like representations are used both for declarative and procedural knowledge and activation spreads over both parts. Thus, it is only active rules that are matched against active facts. Multiple rules fire in parallel, but when there are too many selected rules, only the most active ones are fired. Important differences from ACT* and PUPS are that spreading activation in PI is a side effect of production firing and the presence of an activation-tracing mechanism (i.e., a mechanism keeping trace of the paths the activation has passed). These features of Pi's spreading activation mechanism do no allow it to be regarded as a connectionist approach; it fits better within the marker-passing paradigm. Thai is why I do not regard PI as a hybrid architecture.

The architecture presented in this chapter has some features in common both with ACT* (PUPS) and PI, and others not present in any of them. Thus, the associative mechanism, being an automatic spreading activation process, is closer to Anderson's version, but in contrast to the latter, it performs also a constraint satisfaction task using inhibitory as well as excitatory links. On the other hand, parallel running processes in our architecture correspond to parallel firing of productions in PI and the spread of activation concerns both declarative and procedural knowledge, similarly to PI.

Another architecture which, though not hybrid, is related to ours is NETL (Fahlman, 1979). In NETL, as in our cognitive architecture, many symbolic processors run-in parallel performing a marker-passing task. However, there are important differences. First, in NETL all symbolic processors are the same computing machines, whereas in our architecture they are specialized..ones (i.e., besides some general abilities for marker passing, each processor has some hard-wired knowledge about a specific task: how to perform a specific action of the reasoner). Second, markers in NETL are, actually, marker bits (tokens) each with a specific meaning, whereas markers in the present architecture are essentially pointers to other nodes.

1.2. Models of Analogy: General Approach

7.2.1. Syntactic vs. Pragmatic Approach. In regards to the well-known discussion about the priority of syntactic and pragmatic constraints on analogy (Centner, 1983, 1989; Holyoak & Thagard 1989a, 1989b), I recognize the importance of both constraints and involve both of them in AMBR. In some sense, however, I can be regarded as a stronger supporter of the pragmatic approach than Holyoak as I back up the dominance of pragmatics on all other processes and claim that even the computation of semantic similarity is influenced by pragmatic factors. On the other hand, I have proposed a weaker pragmatic constraint offering a broader understanding of pragmatics as relevance to the whole context—associative relevance—not just to the problem goal.

7.2.2. *Parallel vs. Sequential Processing.* Traditional models of analogy assume that retrieval, mapping, transfer, evaluation, and learning are sequential steps of the reasoning process and even try to model them separately (Centner, 1983, 1989; Hall, 1989; Holyoak & Thagard, 1989a, 1989b; Thagardet al., 1990; Wolstencroft, 1989). On the contrary, I consider them as parallel running processes which influence each other's behavior and therefore cannot be modeled separately. Eskridge (this volume) also advocates the mutual interaction of these processes in proposing his Continuous Analogical Reasoning theory.

7.2.3. Dynamic Aspects and Context Dependence of Human Reasoning. Most models of analogy restrict the context to the target problem's description. Eskridge (this volume) extends it with the type of reasoner's task: problem solving, learning, etc. This, however, cannot account for the dynamic aspects of human reasoning.

Hofstadter (1985) and Mitchell and Hofstadter (1990) deal with these dynamic aspects by allowing two concepts to be considered as similar in one situation and dissimilar in another. This is, however, due to random factors in their model rather than to differences in the contexts (they consider only the problem description).

In the following subsections I will review the models proposed for retrieval and mapping in more detail.

7.3. Retrieval

There are several main approaches to retrieval.

In *case-based reasoning* (Carbonell, 1983, 1986; Kolodner & Simpson, 1986, 1989; Hammond, 1989, 1990) retrieval is performed on the basis of a specific organization of LTM around an indexing scheme. Thus, Carbonell (1986) indexes the potential sources of analogy by the first reasoning steps in the problem-solving activity of that case (he called this derivational analogy). Kolodner and Simpson (1986) use a specific organization of previous cases organized around a generalized episode (MOP, EMOP) and indexed by their differentiating features (this organization could be considered as a discrimination net). Hammond (1989, 1990) indexes old plans by goals and the problems that they avoid and in this way the basic organization of the plan memory is a discrimination net.

In this way each particular model has its specific memory organization (specific indexing scheme) depending on the task it solves. This can be useful for designing systems that run effectively, but it is not psychologically plausible. Memory organization should be flexible enough to cover a wide diversity of problems. So, in my view, static indexing is not an appropriate approach to retrieval.

In *memory-based reasoning* (Stanfill & Waltz, 1986) retrieval is performed on the basis of a general measure of similarity between cases. The representation is very simple: only feature vectors are used. An explicit weighted feature metric is defined, which is then used for parallel evaluation of the distance between the target case and all other cases in the KB and this is how the best match is found. This implies usage of a metric which will be the same for all problems and thus can be applied only for a restricted predefined class of problems.

A hybrid model, *ARCS,* of analog retrieval has been proposed in (Thagard et al., 1990). First, LTM is searched for descriptions containing predicates semantically similar to those in the target. This is a symbolic process performed sequentially for each predicate in the target. Indices are used during the search in LTM. Second, a neural net is constructed (by a symbolic process in a sequential way) which reflects the structural, semantic and pragmatic constraints on *mapping,* with the sole exception of not comparing semantically dissimilarpredicates.

This two-step process is termed retrieval because of the assumption that there is a single mapping which works on the source provided by the retrieval process. If, however, we allow several mappings to run in parallel, then the process can be considered as a sequence of retrieval and mapping. In this case the first "pure retrieval" step is not well grounded: (a) a search is conducted for *all* situations where at least one similar predicate is used; and (b) semantic similarity is restricted to immediate associates only.

The use of *spreading activation* presents another approach to retrieval (Anderson, 1983, 1984; Anderson & Thompson, 1989; Holland et al., 1986; Holyoak & Thagard, 1989). The main idea here is to start by activating the elements of the target description and allowing this activation to spread through the LTM. Thus, some elements become more active then the others and are considered as more plausible sources of analogy. There are a lot of variations of the spreading activation mechanism differing in the way they run and in their use.

Analogy in PUPS "is an action that can be called on the right-hand side of a production" (Anderson & Thompson, 1989). In this case its left-hand side has to be matched against the currently active structures. If several structures match successfully then the conflict resolution strategy chooses the most active one. Spreading activation in ACT* and PUPS is an automatic process which starts in each new memory state (when, for example, a production has fired and put a new source of activation in the working memory) and ends when a stable state is reached (the asymptotic level).

Spreading activation in PI is a discrete one-step process resulting as a side effect of a rule firing. That is why search in PI is much narrower than in ACT* or PUPS. In PI analogy is triggered when the activation of some node exceeds a given threshold.

The model presented in this chapter also uses the notion of spreading activation. The proposed variation is in some aspects closer to Anderson's ideas (it has an automatic nature) and in others to Holyoak and Thagard's work (the analogical mapping is triggered by the change of the focus, that is, by the change of a node's activation). What is different from both approaches is that spreading activation is considered neither as an instantaneous operation before some matching processes start, nor as a side effect of the reasoning itself. Instead, this process runs continuously and in parallel to all other reasoning processes and thus influences their work, reflecting the changing environment.

7.4. **Mapping**

The foremost problem with mapping, how to overcome the combinatorial explosion which will take place in case of an exhaustive comparison between all possibie correspondences, is solved in various ways by various researchers.

Gentner (1983) and Falkenhainer, Forbus, and Centner (1986) in their structure mapping theory (SMT) restricted possible element correspondences only to identical relations (attributes are discarded, objects are put in correspondence after the best mapping is found) and use a purely syntactic criterion for preference—the *systematicity principle:* it is the higher-order relational structure that determines which of two possible matches is made, preferring systems of predicates that contain higher-order relations (a syntactic expression of the tacit preference for coherence and inferential power in analogy). There are several problems with this approach. First, there are a lot of cases where different relations have to be put in correspondence (e.g., on(A, B)—support(B, A), on(A, B)—in-touch(A, B), etc.). Second, object attributes are sometimes quite relevant (e.g., it is the attribute hot(plate) that causes the water to be heated). Third, this single criterion could be insufficient if, for example, two or more higher-order relations exist and the corresponding relational structures are equally systematic. Moreover, there are cases where the more systematic relational structure is less relevant and has to be ignored.

Keane (1988) followed a radical pragmatic approach. According to what he proposed, the first stage of the reasoning process finds the critical object(s) of the target's goal (i.e., the object(s) whose manipulation or use is necessary to the achievement of the goal) and their functionally relevant attributes (FRAs), (The FRAs are those attributes which became salient in the context of a particular goal; one possible operationalization of this idea [Kedar-Cabelli, 1988] is the building of an explanation network.) Next, the reasoning process looks for an object with the same or similar FRA in the source domain and the relations which predicate this critical object in the source are mapped. Finally, other relations are mapped following the "cause" relations.

This approach wiil also cause problems in several cases. First, as Centner would say, this approach is restricted to problem solving where a clear goal exists. Second, it is not always possible to define a critical object and its FRA in the target situation before knowing the plan for achieving the goal; if, however, the plan is known, such an analogy would be used only for plan repairing. Third, in complex situations it would be necessary to use some kind of structural constraints in order to decide which of the relations predicating the critical object should be mapped.

That is why I follow the approach proposed by Holyoak and Thagard (1989b) and use the structural, semantic, and pragmatic constraints together. However, the mechanism for mapping is extended significantly so that a constraint satisfaction machine similar to ACME will be only a part of the whole mapping process. The mechanisms for construction of the network are elaborated, enabling the handling of symmetric and converse relations, the establishing of correspondence between relations with different number of arguments, etc. Pragmatic importance is defined as the associative relevance of the element and the associative mechanism is used for its computation. The semantic constraint is also elaborated further. The mechanism proposed for establishing semantic correspondences searches for similar pairs in both descriptions in parallel instead of one by one for each element and is not restricted to immediate associates like in ACME.

In contrast to ACME, similarity has a context-dependent nature in AMBR. Mitchell and Hofstadter (1990) also consider similarity as context-dependent allowing concepts to make different slippages in different contexts depending on the relevance of the link between them. However, there are some shortcomings of their approach. First, they allow slippages only between immediate associates (i.e., all possible slippages are predefined by the direct links in the slipnet and are not computed during problem solving like in AMBR). Second, they restrict the context to the problem description, so when solving the same problem for a second time, the same slippages will be considered as relevant. They overcome the second problem by including random factors in their model.

There are several basic problems with mapping which are discussed below.

Selection problem: how certain elements of the description are selected for mapping while others are ignored.

There are several solutions proposed so far:

1. Holland et al. (1986) and Holyoak and Thagard (1988) did this on the basis of the representation—it is assumed that the way of representing the source situation reflects the most relevant information. Unfortunately, this is not always the case.
2. Centner (1983) and Falkenhainer et al. (1986) considered attributes and isolated relations as irrelevant. It seems doubtful that relevance can be expressed in such syntactic criteria.

3. Kedar-Cabelli (1988) elaborated the representation in a way that selects only relevant attributes and adds an explanation network to the representation. This is, of course, a good solution, but it is not always possible.
4. Holyoak and Thagard (1989b) did not make any selection and try to map all elements (except the mechanism for dividing the representation in several parts: initial state, solution, etc., and for allowing the mapping only between elements of the corresponding parts).
5. In AMBR more relevant elements (according to their associative relevance) are preferred in the mapping process and selection is based on this criterion. Selection is also done to some extent prior to mapping: having in mind that our representation allows several frames for one and the same situation, each reflecting a particular point of view, it is clear that the fact of selecting a source for analogy by the retrieval mechanism is a selection among all knowledge about that particular situation.

Object identification problem: how the objects of the target domain paralleling those of the source domain are identified. There are three main approaches:

1. The mapping starts with object identification based on attribute similarity, functional similarity, FRA, categorial information, etc. (Keane, 1988).
2. First, the best global mapping is established and then it is used to set up the corresponding object matches (Falkenhainer et al., 1986; Gemner, 1983, 1989).
3. In AMBR a mechanism similar to ACME (Holyoak & Thagard, 1989b) is used which is a form of relaxation search in a connectionist network. In the case of AMBR where objects have corresponding descriptions (in contrast to ACME where objects are semantically empty) this mechanism makes it possible to establish the correspondences between objects and relations in parallel.

There is a number of differences between the constraint networks in AMBR and in ACME. First, in contrast to ACME, in AMBR this is not a separate network but is a temporary built extension of the LTM of the reasoner. Consequently, the state of the reasoner's mind (the presently active elements of LTM and the relations between them) will influence the relaxation search. Second, in contrast to ACME, prior to relaxing the network, in the phase of its construction, it is possible to establish some correspondences (between objects or relations) when they are highly relevant and semantically similar. This will restrict the space of possible mappings described by the network. This makes it possible to model both cases: (a) internal domain or close domain analogies where usually object similarity plays a major role, and (b) abstract analogies between far domains where usually higher order relations dominate the mapping.

8. GENERAL DISCUSSION

8.1. Cognitive Architecture

8.1.1. Continuousness vs. Discreteness Dualism: A Hybrid Approach. A hybrid cognitive architecture is proposed which combines the advantages of a symbolic approach (used for complex structured representation of situations, problems, plans, concepts, etc., as well as for the benefits of a marker-passing mechanism in specialized search tasks) with the strength of connectionism in associative retrieval and soft constraint satisfaction.

These different approaches are highly integrated. It is not the case that each part of the system is organized according to one of these approaches (communicating with the other parts). On the contrary, it is rather the case that different processes (symbolic and connectionist) work on the same structures which are considered as frames by the symbolic processes while the connectionist mechanisms consider them simply as nodes and links. This is possible because of the specific, rather distributed frame organization. Although there is a single frame representing a given object or concept, a lot of frames have to be traversed in order to extract all information about it (i.e., there is a whole network of frames describing an object). This is due to the fact that there is no local information in the frames but only references (pointers) to other frames. The same links are used both for spreading activation and for marker passing, but the two processes are not independent: the possibility for and the speed of marker passing strongly depend on the activation of the nodes.

In this way the connectionist aspect of the architecture continuously "restructures" the knowledge base of the reasoner represented by the symbolic aspect, thus controlling the set of possible inferences at any moment.,,It makes some nodes more accessible and others inaccessible, thereby assigning priorities, restricting the search, etc. This makes the knowledge base dynamic and context-dependent.

There is actually a dual representation of the current situation: (a) an implicit distributed representation—the distribution of activity in the whole network (LTM) according to the associative relevance of each memory element, and (b) an explicit local representation—a structured symbolic representation of the situation including its most important elements and the relations between them.

In this way symbolism and connectionism are considered as dual aspects of human cognition reflecting the fundamental scientific dualism: discreteness vs. continuity.

8.1.2. Reactiveness vs. Inertness Dualism: Parallel Running of Processes. The human reasoner is considered both as reactive and inert; reactive because her behavior continuously reflects the changes in the dynamic environment, and inert because her behavior tends to keep her state of mind constant. The former is demonstrated by the impact of the environment even on highly abstract problem solving. This is modeled in our architecture by a process of automatic retrieval which runs continuously and in parallel with all other processes, thus reflecting the changes in the environment. The latter is demonstrated by the priming effects on human reasoning. It is achieved in our architecture by the particular design of this automatic retrieval process (i.e., a spreading activation over a net with only positive links, a low decay constant, high connectivity of the network, and the emergence of resonance effects).

An important feature of the proposed architecture is the possibility of parallel running of a number of symbolic processes simultaneously with the connectionist spreading activation mechanism. This makes it possible to explain both the possibility to perform several reasoning tasks simultaneously and the interaction between the components of a single reasoning process.

8.2. Model of Analogical Reasoning

A model of human reasoning in problem-solving tasks, called AMBR, is put forth on the basis of this architecture. Its main components are retrieval, mapping, transfer, evaluation, and learning. All of them are composed of the basic architectural constructs: the associative mechanism, the marker-passing mechanism, and other symbolic processes.

Since the purpose of the current research is to build a *cognitive* model, this implies the necessity of an explanation of the impressive flexibility and versatility shown by humans. That is why we do not restrict our consideration to a narrower field: modeling an isolated phenomenon or a particular task and domain. The model might seem too complicated to ensure that AMBR will perform satisfactorily—it depends on too many factors, such as whether a source analog is found or whether a good mapping is established. I think, however, that an adequate model of human reasoning should explain human behavior as it is, including human failures, rather than aim at successfully solving all problems.

For this reason a comparison between a computer simulation of the model and the corresponding results obtained by psychological experiments has been made. It has been shown that the critical points of AMBR reflect human difficulties in problem solving (the same failures are encountered). The computer simulation demonstrates also the same priming effects on problem solving as observed in the corresponding psychological experiments.

An important feature of AMBR is that it demonstrates a dynamic and context-dependent behavior. The latter is influenced both by random fluctuations in the environment and by the memory state of the simulation system thus modeling the dynamic aspects of human reasoning.

8.3. Conscious vs. Unconscious Processing

We can speculate a little on the problems of conscious vs. unconscious processing. My assumption is that only the process corresponding to the focus is controlled consciously and its results are consciously accessible. Since the associative mechanisms runs in parallel with all other processes, it is possible that the focus changes before the conscious process has terminated. In such a case the latter continues at a subconscious level with all possibility of returning back to the conscious level. On the other hand, a process running at the subconscious level can suddenly be activated additionally and in this way continue at ihe conscious level. A subconscious process can activate itself when reaching success. In this way "insight effects" can be demonstrated.

8.4. A Unified Approach to Different Types of Reasoning

AMBR has been initially proposed as a unified computational model of human reasoning in a problem-solving task. Its components can be used for deduction, generalization, and analogy. Thus, for example, schema instantiation or unification in deduction, and schema generalization are performed by the same mapping process as in analogy. Analogy, generalization, and deduction differ in the intermediate results produced by these processes but not in the underlying mechanisms themselves.

In particular, a simple example of this has been demonstrated in the simulation experiment where AMBR generally performs analogical reasoning, but in a specific situation, when the more general rule "IF you put a wooden vessel in the fire THEN it will burn out and the water will disappear" is retrieved by the associative mechanism, a deduction is made.

Psychological experiments (Kokinov, 1990) demonstrate similar priming effects (both near and far) and analogous functions of decrease of those effects in ail three cases: deduction, generalization, and analogy. Other experimental evidence for the hypothesis about uniformity of human reasoning is being searched for.

8.5. Perspectives of Future Work

8.5.1. Theory. As it is stated in the introduction, hybrid models help us to identify two different and complementary aspects of human reasoning, namely the discrete and continuous aspects of its nature. Having recognized them, we can search for a single theory that can explain both of these aspects of human reasoning. The next step along this way could be the investigation of the possibilities for modeling symbol representation and manipulation on the basis °f a connectionist network. This includes connectionist implementations of marker passing (Dyer's signature activations are an example of research in this direction) as well as learning and performing other specialized symbolic processes in a connectionist type network (e.g., considering specialized symbolic processors as specific subnetworks).

8.5.2. Simulation Experiments. I am fully aware of the need for further experimentation in order to test and improve the model. I intend to test how the model scales up and I plan to enlarge the computer simulation by extending the knowledge base, formulating and solving new types of problems, and producing simulations in other subject domains.

8.5.3. Psychological Experiments. I intend to continue the psychological experiments as well. An experimental verification of all components of AMBR is needed, especially a test for mapping and transfer which will give us some important information. An experiment on the evaluation process in analogy, generalization, and deduction has been performed which has demonstrated a dependence of the certainty of reasoning results on the goodness of mapping. The results of this experiment are reported in (Kokinov, 1992).

8.5.4. Predictions. The model makes a number of predictions which have to be tested experimentally.

1. It claims that semantic similarity is context-dependent, so human estimation for the similarity between two concepts should depend on the subjects' preliminary setting, and similarity can happen to be a nontransitive relation for this reason.
2. Since it is claimed that the mapping process heavily depends on the associative relevance of the elements of both mapped descriptions, then it should be possible to demonstrate that in an ambiguous situation, where more than one possible mapping between the situations exists, the subjects' preliminary setting will influence the results of the mapping process.
3. Since it is claimed that the automatic retrieval process is running continuously and in parallel with other reasoning process, it should be possible to demonstrate that the changes in the environment influence the way a problem is being solved. In particular, it should be possible to demonstrate that the appearance of a stone (or a picture of a stone) while solving the "wooden vessel problem" would tend to produce a solution including the heating of a stone instead of the knife or axe, as predicted by the simulation.

REFERENCES

Anderson. J. (1983).ГЛе *architecture of cognition.* Cambridge. MA: Harvard University Press.

Anderson, J. (1984). Spreading activation. In Anderson & Kosslyn (Eds.), *Tutorials in learning and memory.* San Francisco, CA: W.H. Freeman.

Anderson, J.R., & Thompson, R. (1989). Use of analogy in a production system architecture. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning.* New %rk: Cambridge University Press.

Barnden, J., & Pollack, J. (Eds.). (1991). *High-level connections! models. Vol. 1: Advances in connectionist and neural computation theory.* Norwood, NJ: Ablex.

Barnev, P., & Kokinov, B. (1987). Knowledge representation in human memory: An information model. In *Mathematics and education in mathematics.* Sofia: BAS Press.

Burstein, M., & Collins A. (1988). Modeling a theory of human plausible reasoning. In T. O'Shea & V Sgurev (Eds.), *Artificial intelligence (Vol. Hi).* Amsterdam: Elsevier.

Carbonell, J. (1983). Learning by analogy: Formulating and generalizing plans from past experience. In R. Michalski, J. Carbonell, & T. Mitchell (Eds.), *Machine learning.* Palo Alto, CA: Tioga.

Carbonell, J. (1986). Derivational analogy: A theory of reconstructive problem solving and experience acquisition. In R. Michalski, J. Carbonell, & T. Mitchell (Eds.). *Machine learning (Vol. II).* Los Altos, CA: Morgan-Kaufman.

Carbonell, J.G., Gil, Y, Joseph. R., Knoblock, C., Minton, S., & Veloso, M. (1990). Designing an integrated architecture: The PRODIGY view. *Proceedings of the !2th Annual Conference of the Cognitive Science Society.* Hillsdale, NJ: Erlbaum.

Clement, J. (1988). Observed methods for generating analogies in scientific problem solving. *Cognitive Science, 12,* 563-586.

Collins, A., & Michalski R. (1989). The logic of plausible reasoning: A core theory. *Cognitive Science, 13,* 1-49.

De Groot, A. (1983). The range of automatic spreading activation in word priming. *Journal of Verbal Learning and Verbal Behavior, 22,* 417-436.

DeJong, G. (1989). The role of explanation in analogy; or, The curse of an alluring name. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning.* New York: Cambridge University Press.

Duncker, K. (1945). On problem solving. *Psychological Monographs, 58* (Whole No. 270).

Dyer, M. (1991). Symbolic neuroengineering for natural language processing: A multilevel approach. In J. Barnden & J. Pollack (Eds.), *High-level connectionist models, (Vol. I): Advances in connectionist and neural computation theory.* Norwood, NJ: Ablex.

Eliot, L. (1986, Summer). Analogical problem solving and expert systems. *IEEE Expert,* pp. 17-28.

Fahlman, S. (1979). *A system for representing and using real-world knowledge.* Cambridge, MA: MIT Press.

Falkenhainer, B., Forbus, K., & Gentner, D. (1986). *The structure mapping engine. National Artificial Intelligence Conference (AAAJ-86).* Los Altos, CA: Morgan-Kaufman.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science,* 7(2), 155-170.

Gentner, D. (1988). Analogical inference and analogical access. In A. Prieditis (Ed.), *Analogical.* London: Pitman.

Gentner, D. (1989) The mechanisms of analogical learning. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning.* New York: Cambridge University Press.

Centner, D., & Landers, R. (1985). Analogical reminding: A good match is hard to find. *Proceedings of the International Conference on Systems, Man, and Cybernetics.* Tucson, AZ.

Centner, D., & Toupin, C. (1986). Systematicity and surface similarity in the development of analogy. *Cognitive Science, 10,* 277-300.

Gick, M., & Holyoak, K. (1980). Analogical problem solving. *Cognitive Psychology,* 72(80), 306-356.

Gick, M., & Holyoak, K. (1983). Scheme induction an,d analogical transfer. *Cognitive Psychology, 1S(\),* 1-38.

Gilovich, T. (1981). Seeing !he past in the present: The effect of associations to familiarize events on judgments and decisions. *Journal of Personality and Social Psychology, 40,* 797-808.

Hadamard, J. (1945). *An essay on the psychology of invention in the mathematical field.* Princeton, NJ: Princeton University Press.

Hall, R. (1989). Computational approaches to analogical reasoning: A comparative analysis. *Arlificial Intelligence, 39,* 39-120.

Hammond, K. (1989). *Case-based planning.* Boston: Academic Press.

Hammond, K. (1990). Case-based planning: A framework for planning from experience. *Cognitive Science. 14,* 385-443.

Hendler, J. (1989a). Marker passing over microfeatures: Towards a hybrid symbolic/connectionist model. *Cognitive Science, 13,* 79-106.

Hendler, J. (1989b). Hybrid systems (symbolic/connectionist) [Special Issue]. *Connection Science,* /(3).

Hendler, J. (1991). Developing hybrid symbolic/connectionist models. In J. Barnden & J. Pollack (Eds.), *High-level connectionist models (Vol. 1): Advances in conneclionist and neural computation theory.* Norwood, NJ: Ablex.

Hofstadter, D. (1985). Analogies and roles in human and machine thinking. *Metamagical themas.* New \brk: Basic Books.

Holland, J., Holyoak, K., Nisbett. R., & Thagard, P.R. (1986). *Induction.* Cambridge, MA: MIT Press.

Holyoak, K., & Koh, K. (1987). Surface and structural similarity in analogical transfer. *Memory and Cognition. 15(4),* 332-340.

Holyoak, K., & Thagard P. (1989a). A computational model of analogical problem solving. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning.* New York: Cambridge University Press.

Holyoak, K., & Thagard, P. (1989b). Analogical mapping by constraint satisfaction. *Cognitive Science, 13,* 295-355.

Keane, M. (1988). *Analogical problem solving.* Chichester, UK: Ellis Horwood Limited.

Kedar-Cabelli, S. (1988). Toward a computational model of purpose-directed analogy. In A. Priedtis (Ed.). *Analogica.* London: Pitman.

Kokinov, B. (1988a). Computational model of knowledge organization and reasoning in human memory. *Mathematics and Education in Mathematics.* Sofia: BAS Press.

Kokinov, B. (1988b). Associative memory-based reasoning: How to represent and retrieve cases. In T. O'Shea & V Sgurev (Eds.), *Artificial intelligence (Vol. Ш)·.* Amsterdam: Elsevier.

Kokinov, B. (1989). About modeling some aspects of human memory. In F. Klix, N. Streitz, Y. Waern, & H. Wandke (Eds.), *MACINTER II.* Amsterdam: North-Holland.

Kokinov, B. (1990). Associative memory-based reasoning: Some experimental results. . - *Proceedings of the 12th Annual Conference of the Cognitive Science Society.* Hillsdale, NJ: Erlbaum.

Kokinov, B. (1992). Inference evaluation in deductive, inductive and analogical reasoning. *Proceedings of the 14th Annual Conference of the Cognitive Science Society.* Hillsdale, NJ: Erlbaum.

Kokinov, B., & Nikolov, V (1989). Associative memory-based reasoning: A computer simulation. In I. Plander (Ed.), *Artificial intelligence and information-control systems of robots-89.* Amsterdam: North-Holland.

Kolodner, J., & Simpson, R. (1986). Problem solving and experience. In J. Kolodner & C. Riesbeck (Eds.), *Experience, memory, and reasoning.* Hillsdale, NJ: Erlbaum.

Kolodner, J., & Simpson, R. (1989). The MEDIATOR: Analysis of an early case-based problem solver. *Cognitive Science, 13,* 507-549.

Laird, J.E., Newell, A., & Rosenbloom, P.S. (1987). Soar: An architecture for general intelligence. *Artificial Intelligence, 33,* 1-64.

Lange, T., & Dyer, M. (1989). High-level inferencing in a connectionist network. *Connection Science,* /,181-217.

Lange, T., Melz, E., Wharton, C., & Holyoak, K. (1990). Analogical retrieval within a hybrid spreading-activation network. In D. Tburetzky, J. Elman, T. Sejnowski, & G. Hinton (Eds.), *Connectionist models; Proceedings of the 1990 Summer School.* San Mateo, CA: Morgan Kaufmann.

Lewis, R., Huffman, S., John, B, Laird, J., Lehman, *I,* Newell, A., Rosenbloom, P., Simon, T., & Tessler, S. (1990). Soar as a unified theory of cognition. *Proceedings of the 12th Annual Conference of the Cognitive Science Society.* Hillsdale, NJ: Erlbaum.

Michalski, R. (1986). Understanding the nature of learning: Issues and research directions. In R. Michalski, J. Carbonell, & T. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (Vol. 2). Los Altos, CA: Morgan-Kaufman.

Michalski, R. (1989). Two-tiered concept meaning, inferential matching, and conceptual cohesiveness. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning.* New \brk: Cambridge University Press.

Miller, G. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review, 63,* 81-97.

Minsky, M. (1975). A framework for representing knowledge. In P. Winston (Ed.), *The psychology of computer vision.* New York: McGraw-Hill.

Minsky, M. (1986). *The society of mind.* New York: Simon & Schuster.

Mitchell, M., & Hofstadter, D. (1990). The emergence of understanding in a computer model of concepts and analogy-making. *Physica D, 42,* 322-334.

Newell, A. (1990). *Unified theories of cognition.* Cambridge, MA: Harvard University Press.

Palmer, S. (1989). Levels of description in information-processing theories of analogy. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning.* New \fork: Cambridge University Press.

°°lya. G. (1954). *Mathematics and plausible reasoning, vol. I: Induction and analogy in mathematics.* Princeton, NJ: Princeton University Press.

P°lya. G. (1957). *How to solve it.* Princeton, NJ: Princeton University Press.

Posner, M., & Snyder, C. (1975). Attention and cognitive control. In R. Solso (Ed.), *Information processing and cognition: The loyola symposium.* Hillsdale. NJ: Erlbaum.