International Journal of Hybrid Intelligent Systems 6 (2009) 189–209
DOI 10.3233/HIS-2009-0096
IOS Press

Combinations of case-based reasoning with other intelligent methods

Jim Prentzas[a,b] and Ioannis Hatzilygeroudis[a,*]
[a]*University of Patras, School of Engineering, Dept of Computer Engineering & Informatics, 26500 Patras, Greece*
[b]*Democritus University of Thrace, School of Education Sciences, Dept of Education Sciences in Pre-School Age, 68100, Alexandroupolis, Greece*

Abstract. Case-based reasoning is a popular approach used in intelligent systems. It is particularly useful in domains where an abundant number of past cases is available. Cases encompass knowledge accumulated from specific (specialized) situations. Whenever a new case has to be dealt with, the most similar cases are retrieved from the case base and their encompassed knowledge is exploited in the current situation. Combinations of case-based reasoning with other intelligent methods have been explored deriving effective knowledge representation schemes. Although some types of combinations have been mostly explored, other types have not been thoroughly investigated. In this paper, we briefly outline popular case-based reasoning combinations. More specifically, we focus on combinations of case-based reasoning with rule-based reasoning, soft computing methods (i.e., fuzzy methods, neural networks, genetic algorithms) and ontologies. We illustrate basic types of such combinations and also point out future directions.

Keywords: Case-based reasoning integrations, hybrid case-based reasoning, case-based reasoning combinations, hybrid intelligent systems, integrated intelligent systems, case-based reasoning

1. Introduction

The combination of (two or more) different problem solving and knowledge representation methods is a very active research area in Artificial Intelligence. The aim is to create combined formalisms that benefit from each of their components. It is generally believed that complex problems are easier to solve with combined or integrated approaches. The effectiveness of various combined or integrated approaches has been demonstrated in a number of application areas. Popular integrations are neuro-symbolic approaches combining symbolic representations with neural networks [7, 42], neuro-fuzzy approaches combining fuzzy logic and neural networks [83], approaches combining neural networks with genetic algorithms [2], approaches combining fuzzy or neuro-fuzzy systems with genetic algorithms [2] and approaches combining case-based reasoning with other intelligent methods [76,77,96,97].

Case-based reasoning (CBR) resorts to stored past cases to handle new cases [1,28,59,64,119]. The worthwhile experience learned when reasoning with new cases is retained in the CBR system to continuously enhance its effectiveness. Therefore, a CBR system simultaneously performs learning and reasoning [59]. Such an approach is justified by the general notion that in various application fields, people draw on their experience to deal with new incidents. The application of experience to new incident handling is the hallmark of CBR [119]. CBR is becoming an increasingly important AI approach in domains with available (or obtainable) cases. Improvements in data interchange standards, information systems and data entry technologies have produced (and continuously produce) a large number of cases in electronic format [79]. CBR tools are also available to facilitate design and development of CBR systems. Such developments facilitate imple-

*Corresponding author: Deptartment of Computer Engineering & Informatics, University of Patras, 26500 Patras, Greece. Tel.: +302610996937; Fax: +302610960321; E-mail: ihatz@ceid.upatras.gr.

mentation of CBR systems in several domains such as medicine (e.g. [10,107]) in which the number of stored health records constantly increases.

Combinations of CBR with other intelligent methods may be pursued in various domains when the combined system offers advantages in knowledge representation and reasoning compared to each of the combined methods working alone. Generally speaking, almost every intelligent method has advantages as well as certain disadvantages (limitations or challenge issues). Certain intelligent methods have advantages and disadvantages, which are proved to be complementary to some degree. So, it is justified and useful to explore combinations of such methods in order to produce effective combined approaches. The main goal of such combined approaches is to surpass the disadvantages or limitations of each component method and simultaneously benefit from the advantages of each method.

CBR can be effectively combined with other intelligent methods. Roughly speaking, two main trends for CBR combinations can be discerned. One trend involves embedded approaches in which the primary intelligent method embeds one or more other intelligent methods to assist its internal online and offline tasks. The most usual combinations following this trend concern use of other intelligent methods to assist various CBR tasks. The second combination trend involves approaches in which the problem solving process can be decomposed into subprocesses (tasks or stages) for which different representation formalisms are required or available. In such situations, a CBR system as a whole (with its possible internal modules) is integrated 'externally' with other intelligent systems in order to create an improved overall system. Different types of such combinations can be developed.

Popular CBR combinations involve combinations with rule-based reasoning (RBR), model-based reasoning (MBR) and soft computing methods (i.e., fuzzy methods, neural networks, probabilistic reasoning and genetic algorithms). CBR has also been combined with other intelligent methods (e.g. ontologies). Generally speaking, CBR can be combined with another intelligent method according to any of the aforementioned combination trends. In certain CBR combinations (e.g. combinations with RBR and MBR) both combination trends have been followed. In other combinations (e.g. CBR combinations with soft computing) one of the two trends seems to be the most explored one.

According to [118], CBR is not a technology but a methodology that can employ any technique(s) to perform its tasks. Therefore according to this view, approaches using other intelligent methods to assist various CBR tasks should not be considered hybrid CBR systems although they can be considered hybrid AI systems. An objective of our paper is to point out that emphasis should be placed on combinations in which a CBR system is integrated 'externally' with other intelligent systems. We believe there is room for extensive research work in this context. Furthermore, approaches in which CBR is embedded within another intelligent method may also be developed. It is very likely that these two research directions (especially the first one) will produce fruitful results.

In this paper, we briefly discuss aspects involving CBR combinations. We focus mainly on intelligent methods with which CBR is usually combined. Our purpose is not to present a complete survey of developed CBR combinations, but to present their key aspects and the potential for future research work. We believe that the discussion included in this paper will increase understanding of the field involving CBR combinations and the ways CBR can be combined with other intelligent methods. In addition, it may lead to development of new (or overlooked) ways of combining CBR with other intelligent methods. Finally, it is a useful guide to developers/designers of such systems.

The structure of the paper is as follows. Section 2 briefly presents CBR focusing on issues serving as background knowledge for the following sections. Section 3 discusses main types of CBR combinations. This section initially presents general background concerning CBR combinations and discusses afterwards in corresponding subsections issues involving combination of CBR with specific intelligent methods. Finally Section 4 concludes.

2. Case-based reasoning

Case-based representations store a set of previous cases with their solutions in the case base using them whenever a similar new case has to be dealt with [1, 28,59,64,119]. Stored cases encompass knowledge accumulated from specific (specialized) situations that proves useful in handling similar new cases. CBR is particularly useful in domains where an abundant number of past cases is available, similar cases recur often and there may be no explicit domain model [59, 64,119]. According to the application domain, cases may be represented in a variety of forms. Representation of cases may be simple (e.g. plain attribute-value cases) or more complex (e.g. hierarchical) employing formalisms such as frames, objects, predicates, semantic networks [59,119] and ontologies. In complex case representations, case attributes may be connected among them and cases may contain functional dependencies [61]. Also abstract cases enable reasoning in different levels of abstraction. CBR tools may be employed for conversion of existing resources into an exploitable case knowledge (e.g. [29]) facilitating access and retrieval to existing information.

CBR, among others, is useful in performing problem-solving tasks such as design, planning and diagnosis and interpretive tasks such as criticism and justification (e.g. demonstrating the rightness of an argument) [64]. It also proves useful as a retrieval tool by providing assistance in decision-making and teaching [64]. Studies in human reasoning and real-life experience demonstrate the usefulness of reasoning from prior cases in various real-life contexts such as medicine, legal reasoning, weather forecasting based on previous weather records, determining house prices by exploiting similar cases from other real estates [109], learning to use computer programs, learning programming languages and mathematical problem solving [64].

Whenever, a new input case has to be dealt with, the case-based system performs inference in four phases known as the CBR cycle [1]: (i) *retrieve*, (ii) *reuse*, (iii) *revise* and (iv) *retain*. The retrieval phase retrieves from the case base the most relevant stored case(s) to the new case. *Indexing* schemes and *similarity* metrics are used for this purpose. Indexing enables the efficient retrieval of relevant cases from the case base, thus limiting the search time. Indexing defines the circumstances under which a case is useful. The vocabulary used to describe and index cases may be domain-specific or hold across domains. Similarity metrics assess the relevance of the retrieved cases to the new case. A simple approach to similarity assessment is the *nearest neighbor matching* [59]. Weights may be assigned to case features to denote feature importance in similarity assessment. Indexing and similarity depend on case representation. Retrieval involves partial matching since in general there is no existing case exactly matching the case at hand [119]. Retrieval concerns several procedures such as assessing the new situation (i.e. *situation assessment*), *matching procedures* that return a set of partially matching cases and *ranking procedures* that determine the most useful cases from the retrieved ones [59]. Preferences, exclusion criteria as well as other numeric and heuristic procedures may be employed in selecting the most useful cases [59]. One or more retrieved cases, deemed most useful in handling the new case, are used for dealing with it. In the reuse phase, a solution for the new case is created based on the retrieved most relevant case(s). The revise phase validates the correctness of the proposed solution, perhaps with the intervention of the user. Finally, the retain phase decides whether the knowledge learned from the solution of the new case is important enough to be incorporated into the system. *Adapting* the most relevant retrieved case to meet the requirements of the new case is an important process due to the fact that retrieval involves partial matching. Adaptation focuses on differences between the most relevant case and the new case. Various adaptation methods have been developed such as *substitution*, *transformation* and *derivational replay* [59,80]. Adaptation methods require domain-independent and/or domain-dependent knowledge [80]. *Maintenance* is also an important aspect in CBR due to the fact that case base size increases and CBR tasks/environments change over time. Maintenance strategies monitor the CBR system and determine whether, when and how to update system knowledge in order to sustain and improve system performance [28,121].

The development and maintenance of CBR systems requires less knowledge engineering effort compared to RBR systems. It has been demonstrated that in appropriate domains, CBR systems can be prototyped and built faster than RBR systems [64]. The underlying ideas of CBR can be applied consistently across domains. However, the specific implementation of the CBR methods is highly customized to the specific application domain and the CBR designer has to decide from a range of different methods [29]. Therefore, knowledge engineering issues to be faced when developing CBR systems involves among others, case representation, organization of indexing, development/selection of appropriate retrieval, adaptation and maintenance methods.

Advantages of case-based representations [96] involve among others the following:

- *Naturalness of representation*: cases are very comprehensible to the user.
- *Ability to express specialized knowledge:* Cases involve specific situations with precise and well-defined terms.
- *Modularity*: Each case is a discrete, independent knowledge unit that can be easily inserted into or removed from the case base.
- *Easy knowledge acquisition:* In most domains, cases are either available or easily obtainable. Furthermore, it is usually easy to elicit cases from experts [119].

- *No explicit domain model is required:* In certain application domains, an explicit domain model is not available and may be difficult to obtain due to unavailability of expert(s) and/or domain complexity. CBR surpasses this problem.
- *Self-updatability:* Incorporation of knowledge during their operation is an advantage of CBR systems compared to intelligent systems employing other representations. In this way, the reasoning capabilities of CBR systems are enhanced during their operation.

However, there are also issues of CBR that may give problems [96]. Such issues are the following:

- *Inference efficiency problems regarding retrieval*: Such problems may arise when performing case retrieval in very large case bases. Solutions to this issue involve, among others, proper organization of the case base, appropriate indexing of cases to facilitate their retrieval in relevant situations, insertion of only necessary cases into the case base and reduction of case base size.
- *Adaptation issues:* Adaptation can be a complex and time-consuming task usually requiring domain-dependent knowledge and sometimes user intervention [59,80]. Techniques have been developed to automatically acquire adaptation knowledge [80]. Other techniques decrease the need for adaptation by retrieving cases that are easier to adapt (e.g. by refining indices and similarity assessment) [64].
- *Provision of explanations*: A CBR system can provide some kind of explanations [112]. The simplest (but not always adequate) type of explanation involves displaying the most similar case(s) to the user. However, rarely are given explanations for all aspects of CBR process (e.g. vocabulary, adaptation, similarity assessment and other procedures involved in retrieval) [112].
- *Knowledge acquisition problems*: In certain domains, cases are either unavailable or in a limited (insufficient) amount hindering the CBR process (e.g. [75]).

3. Combinations of CBR with other intelligent methods

CBR has been effectively employed in a number of application fields. However, combinations of CBR with other intelligent methods have been explored for more effective knowledge representation and improvement in handling the problem at hand. CBR can be combined with various intelligent methods, such as RBR, MBR and soft computing methods.

To categorize CBR combinations one could use Medsker's general categorization scheme for integrated intelligent systems [78]. Medsker distinguishes five main combination models: *standalone*, *transformational*, *loose coupling*, *tight coupling* and *fully integrated models*. Distinction between those models is based on the degree of coupling between the integrated components. In standalone models independent components of each approach are developed that do not interact with each other during reasoning. They can be used in parallel to compare the independent solutions providing an opportunity to compare the capabilities of each approach. In the transformational model, a system based on one approach is completely transformed to a system based on the other approach. Transformation is made for various reasons such as better representation of the domain, enhanced inference performance, enhanced maintenance etc. The loose coupling model concerns systems in which there are separate integrated components based on each representation method. Communication between the components is achieved via data files. The tight coupling model concerns systems in which the combined components communicate with each other via memory resident data structures. In fully integrated models, the integrated components share structures and knowledge representations and are virtually indistinguishable.

Underlying categories for some of these models are also defined. Main types of underlying categories for loose and tight coupling models involve *pre-processing*, *post-processing* and *co-processing* models as well as *embedded processing* for tight coupling models. In the first two categories, data processing is sequential. Co-processing refers to bidirectional flow of data between the components enabling an enhanced form of interaction and cooperation between them. In embedded systems, a component based on one approach is the primary problem solver, embedding component(s) based on other representation method(s) to handle its internal reasoning tasks. Not all of these combination models and/or their underlying categories have been thoroughly explored in combinations of CBR with other intelligent methods. The types of combination models that have been applied in CBR combinations depend on the nature of the other intelligent methods combined with CBR. Some combination models are difficult to apply to certain CBR combinations. Obviously, the standalone model can be applied in combinations of CBR with any other intelligent method.

Generally speaking, coupling models are the most typical CBR combination models. Especially, embedded coupling approaches constitute a popular trend. The most prominent combinations following this trend useother intelligent methods to assist various internal online and offline CBR tasks as the implementation of CBR tasks is highly customized to the specific application domain. Primary such tasks involve retrieval and adaptation. For instance, other intelligent methods may assist in choosing appropriate indices for the retrieval of cases relevant to the new case, in performing similarity assessment and/or in applying proper adaptation methods. CBR is a generic methodology for building knowledge-based systems and its internal reasoning tasks can be implemented using a number of techniques [118] as long as the guiding CBR principles are followed. For example, the term 'soft CBR' is used to describe CBR systems employing soft computing methods [20,88]. The reverse approach that is, embedding case-based modules into intelligent systems employing other representations to assist in their internal tasks does not seem to be popular with the exception of genetic algorithms (see Section 3.4). In combinations of CBR with RBR, various coupling approaches have been investigated besides embedded approaches [96]. This holds for combinations of CBR with MBR [76, 77]. However, in coupling combinations of CBR with soft computing methods, embedded approaches seem to be the most thoroughly investigated.

It should be mentioned that, in combined approaches, implementation/acquisition and maintenance of functionalities (and knowledge bases) for each one of the combined modules is required. Furthermore, functionalities for communication among modules should be implemented and maintained as well as.

An aspect of interest is the specification of some criteria for deciding on whether an approach/model combining CBR with other intelligent method(s) is suitable for a specific domain. Such criteria can be the following:

1. *Existence of* (*or ability to acquire/construct*) *necessary knowledge sources concerning the application field that may correspond to each of the combined methods.* For instance, in combinations of CBR with RBR, fuzzy logic, neural networks or ontologies, besides case-based knowledge source(s), existence (or ability to acquire) rule-based domain knowledge, fuzzy domain knowledge, training examples (or trained neural networks) and ontologies respectively is necessary. No such combined approach can be implemented unless all corresponding types of knowledge are available (or obtainable).
2. *Ability to implement or acquire modules corresponding to each combined method in case such modules are not already implemented* (*or obtainable*). This criterion concerns all functionalities required by each combined module.
3. *None of the constituent modules working alone seems sufficient to appropriately respond to a significant part of the encountered situations.* In such a case, each knowledge source represents covers a large part of the domain and may be overlapping and/or be complementary with other available sources. Appropriate response means that the produced output should (more or less) comply with the response of the domain expert to the encountered situations. To assess if this criterion is satisfied, prototypes of the combined modules can be used in parallel to compare the independent solutions (see standalone model mentioned above).
4. *A system corresponding to an intelligent method other than CBR requires experience to work effectively in terms of time-efficiency and accuracy.* This complements the previous criterion.
5. *The problem solving process can be decomposed into subprocesses* (*tasks or stages*) *for which different intelligent methods are suitable or available* (*one of them being CBR*). In such situations, the tasks or stages of the problem solving process should be examined to determine which method fits better in which process. This criterion to a large degree overlaps with the two previous ones.
6. *It is known from literature that CBR combinations have been successfully applied to problems of this domain* (*or a similar domain*). It is always useful to exploit the experience gained from the development of similar systems.

From the above criteria, (1)–(5) is more or less necessary to be fulfilled, whereas (6) is optional.

Although Medsker's categorization scheme is generally acceptable by the research community of hybrid intelligent systems, researchers working towards the integration of CBR with other intelligent methods seem not to refer to it. Such is the case in [76,77] which survey integrations of CBR with other approaches and categorize them into three main categories: (a) approaches in which CBR prevails in the reasoning process whereas the other combined intelligent method assists CBR, (b) approaches in which the role of CBR is to assist the other combined intelligent method and (c) balanced approaches in which the role of the combined approaches is balanced. A disadvantage of such a categorization scheme is the difficulty in defining subcategories for the three categories which inhibits this scheme from covering other aspects of CBR combinations, apart from component dominance. Also in [20,88,109] the focus is on embedding other intelligent methods (i.e. soft computing methods) within CBR. Additionally, in [12, 120] the focus is on embedding fuzzy methods within CBR. Finally, in [21] the focus is on employing neural networks to enhance CBR.

In the following, we discuss main issues involving combinations of CBR with RBR, fuzzy methods, neural networks, genetic algorithms, ontologies and multiple other methods. Finally, we present a summary discussion regarding combination of CBR with such intelligent methods.

3.1. Combinations of CBR with RBR

RBR is one of the most popular knowledge representation and reasoning methods. Rules represent domain knowledge in the form of if-then rules: *if* <*conditions*> *then* <*conclusion*>, where <conditions> represents the conditions and <conclusion> the conclusion of a rule. The conclusion of a rule is derived when the logical function connecting its conditions results to true. The inference engine of a RBR system uses the knowledge encompassed in rules as well as the facts about the problem at hand to draw conclusions. Explanations about drawn conclusions can also be provided. A major advantage of rules is their naturalness. Rules involve natural language concepts and therefore their encompassed knowledge can be understood by humans. Furthermore, they constitute a compact way of representing domain knowledge. Inference steps can be traced and explained.

However, RBR has certain important drawbacks as knowledge representation and reasoning method [96]. Such drawbacks are the following:

– *Difficulty in knowledge acquisition* (*called the 'knowledge acquisition bottleneck'*): It is generally time-consuming to elicit rules from experts which may cause delays in development of RBR systems. It also is difficult to acquire a complete and perfect rule set and deficiencies in rule bases affect reasoning accuracy. Complex domains require a very large number of rules.
– *Difficulty in maintenance of large rule bases*: Large rule bases are difficult to maintain due to possible interdependencies between rules.
– *Brittleness of rules*: It is not possible to draw conclusions in cases of missing values in the input data and in cases of unexpected input values or combinations of them.
– *RBR performs inference from scratch not exploiting problem-solving experience*: Such experience could enhance RBR capabilities in terms of time-efficiency and accuracy by representing useful knowledge not encompassed in rules.
– *An RBR system is not self-updatable*: All changes/updates to the rule base need to be performed by a human.

Combination of RBR with CBR can offer benefits when both rule-based and case-based knowledge sources are available (or obtainable). The advantages of RBR and CBR are complementary to a large degree [96]. More specifically:

– On the one hand, RBR provides rule-based domain knowledge representing general knowledge of the domain, naturalness, a compact representation of knowledge which may be desirable in certain applications, rule-based inference (e.g. classification capabilities) and explanation facilities. Finally, it should be mentioned that classification is required in most of CBR tasks.
– On the other hand, CBR provides naturalness and capability to represent specialized knowledge by exploiting available cases. Knowledge acquisition becomes easier. Reasoning in CBR does not have to be performed from scratch as in RBR. Unexpected or missing inputs can be handled surpassing the brittleness of rules. Learning capabilities based on acquired reasoning experience are provided surpassing a limitation of RBR. Case-based explanation facilities are also offered.

For these reasons, the combination of CBR with RBR has been investigated since the 1980s. In fact, different types of the coupling models have been investigated [96] i.e., *sequential processing*, *co-processing* and *embedded processing*.

In sequential processing, the flow of information (produced by reasoning) between the combined modules is sequential or semi-sequential. It includes approaches in which information necessarily passes sequentially through some or all of the combined components in order to produce the final result (e.g. [31,40, 65,105]).

In co-processing approaches, the combined modules closely interact in producing the final result. Such systems can be distonguished into two types: cooperation-oriented, which give emphasis on cooperation, and reconciliation-oriented, which give emphasis on reconciliation. In the former type, the combined components cooperate with each other (usually by interleaving their reasoning steps) for the production of the final result (e.g. [81,103]). In the latter, each component produces its own conclusion, possibly differing from the conclusion of the other component, and thus a reconciliation process is necessary (e.g. [39,66]).

In embedded processing, CBR systems employ one or more RBR modules to perform tasks of their CBR cycle. Typical CBR cycle tasks performed by rules are retrieval and adaptation. As already mentioned, retrieval concerns several procedures in some of which rule-based knowledge may be useful (e.g. situation assessment, employing preferences, exclusion criteria and heuristic procedures in selecting the most useful cases) [59]. Furthermore, CBR systems usually include domain-specific and/or domain-independent adaptation rules. RBR could also be used in case base maintenance (e.g. case base reduction). RBR can also be used for case generation [89]. RBR can also be used to enhance the explanation mechanism of CBR. RBR systems embedding CBR modules do not seem to exist. Such approaches could be interesting, as far as hybridism is concerned. For instance, a CBR component embedded within RBR could assist in tasks such as conflict resolution by providing accumulated experience. In this way, internal RBR tasks could learn from successes and failures and become more tailored to the specific domain improving RBR performance.

Obviously, standalone approaches can also be developed (e.g. [32,50]). For instance, in [50] RBR and CBR are independently applied for failure analysis of mechanical components. Coincidence between RBR and CBR results leads to validation, whereas contradiction between results of both modules or failure of one module to reach a result means that additional data must be collected (e.g. from laboratory tests). Such approaches are useful since RBR is essential for representing general procedures (as executed by experts), but experience encompassed in CBR is indispensable for process acceleration and conclusion verification [50].

The aforementioned different types of coupling models can be applied when the RBR component involves certainty factor rules or fuzzy rules. For instance in [90], a sequential processing coupling model is presented in which RBR contains rules employing a simplified type of certainty factors.

Combinations of CBR with RBR can be regarded as part of a more general trend in AI [96]. This trend concerns development of approaches/methods integrating or combining domain theory and empirical data (e.g. [34,85,116]). Generally speaking, in such approaches domain theory is often rule-based. More specifically in combinations of CBR with RBR, domain theory is rule-based and empirical data the stored cases.

It should be mentioned that systems combining CBR with RBR according to a non-embedded approach may be used as didactic tools for non-experts by exploiting the complementary knowledge types (i.e. general and specialized) represented by RBR and CBR (e.g. [11, 32,50]).

The research work involving combinations of CBR with RBR could provide guidelines to research work concerning combinations of CBR with intelligent methods other than RBR. It stresses out that various coupling approaches can be investigated. Benefits can be gained from a system combining CBR with another intelligent method according to a non-embedded approach. We use categories defined for combinations of CBR with RBR in following sections.

3.2. Combinations of CBR with fuzzy methods

Fuzzy systems are based on *fuzzy sets* which extend the classical notion of sets. Fuzzy sets employ *membership functions* to express membership degrees of elements into these sets. In fuzzy logic, the degree of truth of a statement can range between [0, 1] and is not constrained to the two truth values {true, false} as in classic binary logic. These notions enable representation of and reasoning with real-world situations involving inherent imprecise concepts. Fuzzy logic employs membership functions for linguistic domain variables used in real-world applications. *Fuzzy expert systems* constitute a popular application of fuzzy logic. In such systems, sets of *fuzzy rules* are used to infer conclusions based on input data. Fuzzy rules include fuzzy variables in conditions and conclusions. A fuzzy rule may replace more than one ‘conventional’ rule reducing the rule base size. Fuzzy logic inference process includes three phases: *fuzzification* of inputs (via membership functions), application of fuzzy rules and *defuzzification* (to produce the output).

Despite the advantages of fuzzy logic in representing and reasoning with imprecise terms, there are also some major limitations to this approach:

- *Knowledge acquisition problems*: Fuzzy systems need to be customized to the specific application. However, there is no inherent mechanism in fuzzy logic for learning from empirical data. Encompassed knowledge is acquired from humans (e.g. experts, system designers). This hinders the definition of various system parameters. For instance it may be difficult to define parameters such as membership functions and fuzzy operators for which the best system performance is achieved. Acquiring fuzzy rules from humans can also be tedious [78]. However, there are alternative methods for acquiring fuzzy rules. When corresponding training data is available, fuzzy rules can be induced from fuzzy decision trees (e.g. [110]) or extracted from trained neural networks (e.g. [15]).
- *Adapting to the environment as new data becomes available is not possible*: Whenever updates to knowledge base items of a fuzzy logic system are deemed necessary (such as changes to membership functions and/or fuzzy rules), they need to be performed by humans.

Such limitations are surpassed in approaches combining fuzzy logic with other intelligent methods (e.g. neuro-fuzzy approaches, approaches combining genetic algorithms with fuzzy or neuro-fuzzy systems).

Combination of fuzzy methods with CBR can thus offer advantages when both types of knowledge sources are available (or obtainable):

- On the one hand, fuzzy methods provide imprecision handling, a significant aspect in many real-world applications. In case that fuzzy RBR is combined with CBR, available general domain knowledge in the form of fuzzy rules is also exploited. Furthermore, fuzzy rules provide a compact representation of knowledge which may be desirable in certain applications. Finally, most of the CBR tasks involve some level of imprecision.
- On the other hand, CBR provides the capability to exploit available case data and to learn based on acquired reasoning experience surpassing a limitation of fuzzy logic. Reasoning in CBR does not have to be performed from scratch as in fuzzy RBR.

It should be mentioned that fuzzy concepts can be effectively incorporated into other methods and techniques. Such examples involve fuzzy clustering, fuzzy similarity functions, fuzzy decision trees, fuzzy rough sets, fuzzy cognitive maps and fuzzy Bayesian networks. We briefly discuss issues concerning combinations of neuro-fuzzy approaches with CBR in Section 3.6.

CBR can be combined with fuzzy methods in fruitful ways. A usual approach is the incorporation of fuzzy concepts into a CBR system in order to improve CBR aspects [12,20,88,109]. Other coupling approaches although less frequent than the aforementioned ones can be implemented as well.

Combinations embedding fuzzy concepts into CBR have been vastly explored, given that imprecision and vagueness are inherent in various CBR tasks [120]. Fuzzy terms may be used in case representation enabling a flexible encoding of case features that encompass imprecise, vague or incomplete information [99]. Fuzzy logic may also prove very useful in indexing and retrieval. Fuzzy indexing enables multiple indexing of a case on a single feature with different degrees of membership [109]. Fuzzy similarity assessment and matching methods can produce more accurate results [100]. For instance, fuzzy k-nearest neighbor, a fuzzy version of k-nearest neighbor, can be used in retrieval [67]. The use of fuzzy sets increases the chance of good match, avoids 'too few' retrieved cases (i.e. avoids excluding 'nearly' similar cases) and enables handling of numeric and non-numeric features with linguistic values [19]. Fuzzy methods guarantee robustness and accuracy in cases of missing and noisy data [104]. Fuzziness can also be employed to measure temporal similarity [53]. Fuzzy clustering methods can also be applied to case retrieval reducing case retrieval time [60]. Also in [123] fuzzy analytical hierarchy process is employed to retrieve relevant cases to the input case and then filter retrieved cases not complying to specific criteria. Fuzzy rough sets can be employed to assist in retrieving reusable cases [101]. In such an approach, retrieval is performed according to reusability of cases to the current application by applying fuzzy rough sets to the history of a case's use in different applications. In [54] the case base is populated with fuzzy rules. Such an approach simplifies knowledge representation and reduces case base size thus minimizing computational complexity in terms of time and memory usage. In addition, fuzzy adaptation rules can be employed in case adaptation enhancing conventional rule-based adaptation knowledge. With fuzzy adaptation rules, less cases need to be retained in the case base [110]. Fuzzy adaptation rules can be induced from cases using fuzzy decision trees [110]. Also in [113] an approach to case base construction is described using fuzzy similarity relations. Such an approach can be useful in domains in which cases are either unavailable or in a limited (insufficient) amount. Fuzzy approaches can be employed in case base maintenance tasks. For instance in [110], fuzzy adaptation rules are used to reduce the case base size by selecting representative cases and removing redundant ones. Fuzzy rough sets can be used for case feature reduction and weighting [51]. When cases have a large number of features it may be necessary to find significant features and delete redundant ones in order to enhance retrieval efficiency. Also fuzzy integrals can be used to determine case base competence [108]. Fuzzy rules generated using rough set theory can be exploited for case generation [87]. In this approach, a case is a cluster granule and involves a reduced number of relevant features. The methodology is suitable for mining data sets, large both in dimension and size, due to its low time requirement in case generation as well as retrieval.

The focus on using fuzzy methods in combination with CBR mainly involves enhancement of CBR aspects. As far as hybridism is concerned, it would be more interesting to develop other coupling approaches in combinations of CBR with fuzzy systems besides ones embedding fuzzy concepts within CBR, above mentioned. For instance in [70] a sequential coupling approach is applied. In this approach, four systems are invoked sequentially with the results of each system passed on to the next one in the sequence: three CBR systems and last a fuzzy system (a fuzzy belief network). Also in [37] a CBR component enhances fuzzy cognitive map techniques. In this approach, whenever the fuzzy cognitive map is unable to infer a decision based on the specific input data, the CBR component is invoked to retrieve the most similar case which is used to update the weights of the fuzzy cognitive map and the decision is made from the updated fuzzy cognitive map.

The work concerning combination of RBR with CBR [96] could potentially be improved with the use of fuzzy rules. Complementarities of fuzzy RBR and CBR (i.e. representation of general and specialized knowledge) can offer advantages. For instance [94] present a combination of a CBR component using fuzzy terms in case representation with a fuzzy RBR component. The combination follows the reconciliation-oriented coupling model. Both modules are invoked in parallel and a type of numeric reconciliation is performed: the similarity value of the most relevant case and the conclusion degree of the fired rule are averaged to produce a more accurate and realistic conclusion degree. Also in [117] a combination of CBR with a fuzzy RBR component is presented. The approach follows the reconciliation-oriented process by invoking independently the CBR and RBR components and combining the corresponding results according to specific formulas. In addition, MARS [30] aggregates evidence from rules and cases through possibilistic reasoning. All knowledge in MARS is represented as possibilistic rules. Cases are first converted into this form using certain knowledge concerning each case, which enables its representation as a rule. In [69], CBR is combined with fuzzy RBR according to the sequential coupling approach (i.e., fuzzy RBR is invoked after CBR). It should also be mentioned that in [54] a standalone approach involving a CBR system and a fuzzy RBR system has been followed prior to the incorporation of fuzzy RBR into CBR. The combined system outperforms each one of the CBR and fuzzy RBR system working alone. These approaches demonstrate that combination of fuzzy RBR with CBR can follow the different coupling models presented in [96]. Such approaches could yield fruitful results and may provide the impetus of exploring different coupling models in combinations of CBR and fuzzy methods.

3.3. Combinations of CBR with neural networks

An artificial neural network is a computational model that tries to simulate biological neural networks [43]. Neural networks are used for performing classification and clustering tasks. A neural network consists of many simple interconnected processing units called *neurons*. Each connection from neuron u_j to neuron u_i is associated with a numerical weight w_{ij} corresponding to the influence of u_j to u_i. The behavior of a neural network is determined by neuron interconnections and neuron parameters. Training data are used to train a neural network to perform its desired function. There are three main types of learning paradigms: *supervised*, *unsupervised* and *reinforcement learning*. Various learning algorithms have been applied to train neural networks. Back propagation is the most well-known such algorithm. Different types of network architectures exist: feedforward networks, recurrent networks and lattice structures [43]. Different types of neural networks have been developed such as back propagation neural networks, Radial Basis Function networks, Self-Organizing Maps, Hopfield networks, Boltzmann machines, ART networks, etc. Neural networks have several advantages [42]. They are able to learn from training data which are available in several applications. The output of a neural network is computed efficiently. Neural networks are able to generalize that is, produce a correct output for input value combinations not present in the training set. Output is also produced in case of incomplete input data.

Neural networks also have a number of disadvantages [42]:

- *Training time and convergence problems*: The required training time may be extensive and convergence to an acceptable solution is not always assured.
- *Initialization issues*: The initialization of weights may play an important role in the training process leading to different solutions.
- *Topology design problems*: Determination of neural network topology (such as finding the required number of hidden nodes) is done on a trial-and-error basis.
- *Incomprehensibility of encompassed knowledge*: It is difficult to comprehend the knowledge encompassed in a neural network. More specifically, it is difficult to associate the weights and nodes of the neural network with specific domain concepts since the knowledge of training examples has been distributed over the whole network. Therefore, a neural network cannot be decomposed into components and form a modular structure. A further negative consequence is the difficulty in transferring the knowledge of a trained neural net to other related application domains. The comprehension of knowledge contained in neural networks can be achieved by knowledge extraction methods [115]. However, the extracted knowledge may not faithfully represent the behaviour of the neural network.
- *Difficulty in providing explanations*: Due to the incomprehensibility of knowledge encompassed in neural networks, it is also difficult to explain reached conclusions.
- *Incremental learning issues*: Incremental learning though desirable since the complete training set may not be available a priori, is not always possible in all types of neural networks. This means that in certain types of neural networks, whenever new training data becomes available, the network needs to be retrained with the training set containing the previous and new training data.
- *Potentially available symbolic domain knowledge is usually not exploited in a direct way*.

Combination of neural networks with CBR can offer advantages when both types of knowledge sources (i.e. training examples and cases) are available:

- On the one hand, neural networks provide efficiency, generalization and robustness that are important aspects in numerous domains. Furthermore, classification and clustering functions are necessary in several CBR tasks.
- On the other hand, CBR offers naturalness, modularity and explanation facilities to the overall system by exploiting available cases. Incremental learning is also inherent in the CBR process. Finally, knowledge can be transferred to closely related application domains.

Neural networks are usually employed by CBR to perform tasks such as indexing, retrieval and adaptation. In this way, appealing characteristics of neural networks such as parallelism, robustness, adaptability, generalization and ability to cope with incomplete input data are exploited [21,109]. Due to the fact that different types of neural networks have been developed, different types of neural capabilities for classification and clustering can be exploited. An aspect of interest is that there can be two different main ways of using neural networks to enhance CBR. More specifically, neural networks may be separate modules of a CBR system or a neural network may be used to build a CBR system [21]. In the second case, a unique structure is derived combining characteristics from both paradigms [95]. For instance, in such an approach, neurons may represent cases [21]. Therefore, very tight coupling schemes are derived.

Neural networks can assist in choosing the proper case representation by analyzing available data [78]. For instance, in [128] a back propagation feedforward neural network is employed to assist in retrieval. The network is trained using a processed form of the initial cases as training data. Clustering capabilities of neural networks (e.g. Self-Organizing Maps) can be employed to improve retrieval by organizing cases to clusters (e.g. [17,131]). In such an approach, retrieval becomes more time-efficient and accurate, as first the nearest cluster for the input case is determined and afterwards most similar cases in corresponding cluster are retrieved. Neural networks can also be employed to produce feature weights [49,68] exploiting cases as training data. It may be useful to embed neural networks with incremental learning capabilities into CBR such as ART or ART-Kohonen neural network [126] that adapt to changing environments. A single neural network can be used to perform different CBR tasks such as indexing, retrieval, adaptation [46]. Certain CBR approaches have employed different types of neural networks for the various internal CBR tasks (e.g. [33,106]). Knowledge extracted from neural networks could also be exploited by CBR [21,109,126]. Neural networks can also be used for case base reduction and other maintenance tasks [78].

A direction that would be interesting to pursue involves non-embedded coupling approaches combining CBR with neural networks. In [56] a standalone comparison between CBR and neural networks is performed and the conclusion is that further research is required to develop a hybrid model integrating the various methods (such as neural networks, CBR and genetic algorithms). In [3] among others, a combination of a neural network and CBR according to the reconciliation coupling approach is presented for accuracy improvement. The case base consists of the neural network training examples and each case is indexed by its real solution and by its neural network solution. The approach can employ any type of neural network performing classification. In [124,125] a Self-Organizing Map is combined with CBR to alleviate overlaps between clusters within each class and improve classification accuracy compared to each method working alone. In [74] an approach combining CBR with an incremental learning neural network (i.e. ARN2) is described. The neural network processes prototypical cases, which constitute the majority, whereas CBR processes particular and boundary cases. First, the neural network component is invoked and, according to its activation state, the new case is processed by the neural network or CBR. In [22] a semi-sequential approach is presented in which the back propagation neural network is first invoked to classify the given input case. If the input case is classified to specific categories, the CBR module is invoked, otherwise reasoning ends.

3.4. Combinations of CBR with genetic algorithms

Genetic algorithms employ evolution techniques to find adequate solutions to problems. Candidate solutions to a problem are represented as strings called chromosomes. Crossover and mutation operators are applied to existing candidate solutions in order to produce new candidate solutions. A function (called the *fitness function*) producing a numerical value is used to evaluate a candidate solution's ability to solve a problem. If this numerical value is below a threshold, the corresponding candidate solution is not retained in the pool of candidate solutions. Genetic algorithms are useful in challenging and complex tasks such as planning, scheduling and resource allocation.

In spite of their advantages, genetic algorithms have also some limitations or challenge issues:

– *Setting certain basic parameters such as population size, crossover rate, mutation rate and initial population.* Genetic algorithms randomly initialize their initial population. Setting these parameters affects performance of genetic algorithms (i.e. convergence time and solution accuracy).
– *Provision of explanations*: A limitation of genetic algorithms involves inability to provide explanations for the derived solution. In this way, it is not quite clear how the genetic algorithm reached to the solution or why the reached solution is profitable. In complex problems, a larger number of operations take place and the genetic algorithm needs explanation support. Such explanations could also assist in revealing knowledge implicit in the genetic algorithm process that could be useful in the future to system designers and developers (e.g. for design of better search strategies).

Combination of genetic algorithms with CBR can thus provide benefits to the overall system:

– On the one hand, genetic algorithms provide techniques for searching problem solutions, optimizing system aspects and adaptability. Furthermore, they provide compact representation of problem parameters as numeric values and representation of a variable number of (not predetermined) possible solutions [78].
– On the other hand, CBR provides the capability to incorporate experience within the system by exploiting past cases and also by continuously learning based on new incident handling. CBR also offers naturalness and explanation facilities. Finally, most of CBR tasks involve some type of search or optimization whereas setting genetic algorithm parameters requires experience (e.g. learning from successes and failures).

CBR can be combined with genetic algorithms in various ways. Usual combinations involve use of genetic algorithms to optimize (one or more) aspects of a CBR system. On the other hand, CBR can be exploited to enhance genetic algorithms. Other types of combinations of CBR with genetic algorithms can be also implemented.

Genetic algorithms can be used within CBR to enhance indexing and retrieval. So, they can be used to assign case feature weights enhancing similarity assessment [18,35,47,129], to perform feature selection [55] and generally to select relevant indices for evolving environments. Genetic algorithms can also be used to retrieve multiple similar cases [127]. If k nearest neighbor retrieval is applied, genetic algorithms can be used to find the optimal k parameter in order to improve the retrieval accuracy [5]. Genetic algorithms can also assist in organizing cases to clusters by optimizing conventional clustering algorithms such as K-means [57]. In such an approach, retrieval is enhanced in terms of time-efficiency and accuracy.

Genetic algorithms can be also used to enhance case reuse/adaptation [45,52,91] by producing more creative solutions and avoiding production of some repetitive solutions. Therefore, genetic algorithms may assist CBR in performing creative reasoning [64].

Additionally, genetic algorithms can be used to optimize case representation e.g. by performing case feature discretization [55] and removing irrelevant features. Such optimizations improve accuracy, search time performance and storage requirements.

Furthermore, genetic algorithms can be used to perform instance selection i.e., finding the representative cases in a case base and determining a reduced subset of a case base. In this way, time performance is improved, by reducing search space, and accuracy may be improved through elimination of noisy and useless cases [5,111]. Genetic algorithms can also be used to initially generate cases for CBR [111] surpassing knowledge acquisition problems in domains where a sufficient amount of cases is not available. It is also quite usual to employ genetic algorithms to simultaneously optimize more than one CBR aspect (e.g. [5, 55]). The use of genetic algorithms to optimize CBR aspects is applicable to CBR maintenance tasks [25], since the effective operation and the enhancement of reasoning/learning capabilities of a CBR system requires evolution of its aspects through time. Such maintenance tasks could involve controlling the quality and size of the case base (e.g. find redundant, overlapping, conflicting cases) [78].

CBR, on the other hand, can be employed to enhance genetic algorithms in different ways. CBR can be used to genetic algorithm parameter tuning. For instance in [92], CBR is used to automatically supply the optimal configuration for the genetic algorithm by setting three basic genetic algorithm parameters: the population size, the crossover rate and the mutation rate parameter. CBR can also be applied to genetic algorithms by creating cases to track the history of a search. This case base can contribute to the understanding of how a solution was reached, why a solution works, and what the search space looks like. It could thus be used to design highly tailored search strategies for future use [72]. Such an approach could therefore be used to explain the results of the genetic algorithm and for knowledge extraction. Moreover, similar stored cases can be also incorporated into a genetic algorithm to reduce convergence time and improve solution accuracy. As already mentioned, genetic algorithms randomly produce their initial population. Instead, relevant stored cases can be used as part of the initial population (solution). Additionally, relevant stored cases can be periodically injected into the pool of chromosomes while the genetic algorithm runs [16,73]. In certain approaches, CBR is exploited by genetic algorithms in both ways (i.e., for explanation-knowledge extraction and case injection) [93].

Other types of combinations of CBR with genetic algorithms can be also implemented. For instance, in [36] the standalone model is applied to a real-world problem. CBR and genetic algorithm approaches are developed, analyzed and compared. The purpose is the improvement on features/tasks of the two individual approaches (i.e., the representation of genetic algorithm and the different phases of CBR) and the study of hybrid systems (e.g. trying to put together the advantages of both techniques in a single system).

In [44] a cooperation-oriented approach in an intelligent e-learning system is presented. A genetic algorithm generates personalized curriculum sequencing, whereas CBR performs summative examination or assessment analysis and also provides capability to support corrective activities and second formative assessment.

In [8] a semi-sequential approach is presented in which the genetic-based component is first invoked and if it is not able to produce a solution within specific time limits, the CBR component is invoked.

3.5. Combinations of CBR with ontologies

An ontology may be used to define a domain and reason about domain properties by formally representing a set of domain concepts and relations between them. An ontology provides a shared vocabulary, which can be used to model a domain. Typical ontology components include individuals (e.g. instances or objects), classes, individual and class attributes, relationships concerning classes and individuals, events, restrictions, rules and axioms. Examples of relationships involve the ‘is-a’ and ‘part-of’ relationships. Ontology languages are use to encode ontologies. Various ontology languages exist. Such ontology languages (and ontology language families) among others are Common Logic [24], Rule Interchange Format (RIF) [102], Knowledge Interchange Format (KIF) [58], Ontology Interchange Language (OIL) [84], DARPA Agent Markup Language (DAML) [26], DAML + OIL [27], which combines features of DAML and OIL, and Web Ontology Language (OWL) [86]. Several of those ontology languages have been developed for the Web and the Semantic Web.

An important strength of ontologies is their information-rich nature [79]. Ontologies facilitate knowledge sharing and reuse. They can provide an explicit conceptualization describing data semantics and a shared and common understanding of the domain knowledge that can be communicated among agents and application systems [14]. Ontologies play a crucial role in enabling the processing and sharing of knowledge between programs on the Web [62]. Intelligent Decision Support Systems in the semantic Web framework should be able to handle, integrate and reason from distributed data and information on the Web [9].

A major drawback of ontologies involves inconsistent completeness in terminologies with respect to structure and content. For instance, in medical ontologies, concept coverage for clinical sub-domains can vary for different termonologies [79]. In addition, there may be different ontologies in the same domain [130] due to different domain perceptions and/or different ontology languages. Merging of different ontologies is a frequently addressed problem. In general, two even very similar ontologies cannot be merged and where two ontologies can be merged special treatment is necessary [23]. Generally speaking, the merging process is a manual process.

Therefore, ontologies can be combined with CBR in various ways. Ontologies can be useful to a CBR system regarding different aspects, as they formalize explicitly declared relevant knowledge [38]. Ontologies can be used as:

– the vocabulary to describe input problems (or queries) and/or cases,
– a knowledge structure where the cases are located,
– the knowledge source to provide semantic reasoning methods for similarity assessment and case adaptation that are reusable across different domains [29].

Ontologies can provide vocabulary for case representation and the ability to represent structured cases [9, 29,63,122]. Ontologies give semantic coherence and structure to cases [38]. Ontologies may assist in handling the synonym problem: the same concept terms may have different meaning in different contexts or different terms may be used to represent the same concept [63]. By providing background knowledge, ontologies can be used to complement the specific knowledge of cases [38]. Ontologies can provide the vocabulary for indexing cases [6,29,122]. Case indices can be represented in the ontology by individuals and the similarity of cases is reduced to the similarity between the individuals [122]. Ontologies may also be used for case abstraction [9].

Ontologies can provide the vocabulary and knowledge representation model to describe or formulate input problems (or queries) [61]. The support provided by ontologies reduces the amount of work required to input information and enforce domain model integrity. Such support involves among others, synonym problem handling [63], inheritance of the components, properties and relations from classes, creation of virtual components corresponding to complex structural relations and model verification [122].

Ontologies may enhance similarity assessment. Structural similarity over ontology can be used in retrieval [38]. Furthermore, with use of ontologies, similarity measures guarantee that (when possible) all query elements are valid elements. Semantic similarity metrics quantify similarity in meaning between two concepts. More specifically, semantic distance measures the relative closeness between two concepts of interest from a terminology or concept-oriented view [79]. A simple ontology-based similarity algorithm computes the similarity between two ontology objects by counting the length of the shortest path connecting them in the ontology hierarchy [79]. Use of ontologies in similarity assessment is practical in large and complex domains such as medicine that are rich in synonymy and semantically similar/related concepts [79]. For example, without use of ontologies the synonym problem may cause the mismatching of similar cases [63].

Ontologies may also be used to assist adaptation methods taking into account the context [9]. For instance in case of substitution, ontologies can provide useful information in order to enable selection of substitute values maintaining dependencies and other relations (e.g. temporal relations [38]). Adaptation rules can also be represented by ontologies [122].

In some approaches, ontologies assist in performing all CBR tasks (i.e., input problem representation, case representation, indexing, similarity assessment, adaptation) [122].

In total, (domain-independent) ontologies can formalize CBR knowledge and assist in implementation of CBR knowledge engineering tools aiming to reuse, flexibility and usability [29]. CBR tools employing ontologies demonstrate benefits since ontologies allow formal specifications that add a precise meaning and enable reasoning support [29]. There are also important benefits regarding reuse because ontologies can be shared by different systems. Such tools may use Semantic Web technologies and enable connection with Description Logic reasoners that work with the ontologies [29]. Such a tool is jCOLIBRI [29] based on CBROnto. CBROnto is an ontology that incorporates reusable CBR knowledge, including terminology plus a library of reusable problem solving methods. CBROnto provides with a general test-bed of CBR methods [38].

3.6. *Combinations of CBR with multiple other intelligent methods*

The previous sections focused on combinations of CBR with an individual other intelligent method. However, intelligent systems have been developed combining CBR with multiple other intelligent methods. Such multi-integrated paradigms usually follow a coupling model.

Obviously, a CBR system may employ multiple intelligent methods (e.g. rules and various soft computing methods) to perform its internal tasks [118]. Typical examples of approaches employing multiple soft computing methods within the CBR cycle are presented in [33,106]. In [33] all four phases of the CBR cycle employ soft computing methods. Employed soft computing methods are a self-organizing neural network for retrieval, a radial basis neural network for reuse, fuzzy systems for revise and all soft computing methods for retain. In [106] fuzzy logic, (supervised and unsupervised) neural networks and a genetic algorithm are employed for case representation, indexing, retrieval and adaptation.

More interesting approaches concern multi-integrated systems not following the embedded approach. Typical such multi-integrated approaches involve combinations of CBR, RBR and MBR (e.g. [82]). Such multi-integrated approaches seem to be effective because combinations of CBR with RBR and MBR individually have been thoroughly investigated. Quite often such systems have been implemented to deal with deficiencies of earlier systems combining CBR with only one of the other two intelligent methods (e.g. RBR or MBR). Multi-integrated approaches combining CBR with other intelligent methods besides RBR/MBR can be developed too. For instance, ontologies could constitute an interesting candidate method that could be combined with CBR and another intelligent method in order to facilitate knowledge sharing and reuse among the integrated system components themselves [13] and among integrated systems. Such a combination could be useful in Web-based systems that need to share knowledge. Fruitful such approaches could involve combinations of CBR, ontologies and RBR/MBR. For instance in [14] an approach combining CBR, RBR and an ontology is presented. Moreover, in [71], an approach in which different subsystems using different representation formalisms is presented. Each representation method is appropriate for specific task(s) in mold-base design. Such methods are CBR, neural networks and a formalism combining rules with frames.

Multi-integrated paradigms could also be considered approaches combining CBR with certain neuro-symbolic or neuro-fuzzy modules. More specifically, we refer to situations in which the neuro-symbolic (neuro-fuzzy) module fully integrates the neural and symbolic (fuzzy) approach.

Such neuro-symbolic approaches aim to benefit (to a lesser or larger degree) from advantages of the combined neural and symbolic methods (e.g. [34,116]). They aim to benefit from advantages of symbolic methods such as naturalness, modularity, exploitation of available symbolic domain knowledge and explanation facilities for reasoning process. The symbolic component is used to surpass drawbacks of neural networks such as difficulty to comprehend their encompassed knowledge, difficulty in explaining reached conclusions, difficulty in defining neural network parameters (e.g. topology, initial weights), time-consuming training process, not assured convergence to global minima and not exploiting potentially available symbolic domain knowledge in training. The neural component is used to surpass potential disadvantages of the symbolic method such as knowledge acquisition problems, problems in drawing conclusions in cases of missing values in the input data and in cases of unexpected input values or combinations of them (i.e. inability to generalize), inference time-efficiency problems, difficulties in maintaining large symbolic rule bases and not exploiting potentially available empirical data in the form of training examples. Although various such neuro-symbolic approaches have been developed, most of them put emphasis on the neural component sacrificing certain advantages of symbolic methods. Therefore, in most such neuro-symbolic approaches it is difficult to comprehend their encompassed knowledge and also difficult to explain reached conclusions. Also depending on the employed neural module, it may not be possible to perform incremental learning when new knowledge (training examples and/or symbolic) becomes available but the neuro-symbolic module should be constructed from scratch based on all available knowledge.

Combination of neuro-symbolic approaches with CBR can thus offer advantages when all required types of knowledge sources are available (or obtainable) that is, symbolic domain knowledge, training examples and cases. Exploiting all such knowledge sources may provide improved knowledge representation to the overall system as each knowledge source may have gaps or imperfections in domain representation. More specifically:

- On the one hand, the neuro-symbolic module provides the advantages of neural networks and (certain) advantages of the symbolic method which are required in several domains. Available domain knowledge (i.e., symbolic, training examples) is also exploited. Furthermore, the neuro-symbolic module provides a compact representation of knowledge and its explanation facilities (if any) which may sometimes be desirable.
- On the other hand, CBR provides the capability to exploit available cases and to perform incremental learning. Finally, CBR can provide naturalness and explanation facilities to the overall system.

Neuro-symbolic modules could be used within CBR instead of plain neural or symbolic components. Non-embedded coupling approaches can be applied as well. For instance, in [41,98] a neuro-symbolic method is combined with CBR according to the reconciliation coupling approach. The neuro-symbolic method involves neurules, a type of hybrid rules fully integrating symbolic rules with the adaline unit. Also in [4] a neuro-symbolic method fully integrating symbolic domain knowledge with neural networks (i.e. knowledge-based neural networks) is combined with CBR according to the reconciliation coupling approach. In all these three approaches, combination of CBR results in improved overall accuracy by exploiting all types of available knowledge sources.

Neuro-fuzzy approaches aim to benefit from advantages of the combined neural and fuzzy methods. The fuzzy component is used to surpass (to a certain degree) limitations of neural networks such as incomprehensibility of encompassed knowledge and difficulty to explain reached conclusions. Furthermore, it enables representation of imprecision and exploitation of available fuzzy domain knowledge. The neural component is used to surpass disadvantages of the fuzzy method such as knowledge elicitation problems by learning from available training examples. A disadvantage of certain neuro-fuzzy approaches is the partial loss of naturalness of fuzzy domain knowledge.

A system combining neuro-fuzzy approaches with CBR can benefit from the exploitation of all types of knowledge sources that is, fuzzy domain knowledge, training examples and cases (with the precondition that they are available or obtainable):

- On the one hand, the neuro-fuzzy module provides advantages of neural networks and certain advantages of fuzzy methods. Available domain knowledge (i.e., fuzzy, training examples) is also exploited.
- On the other hand, CBR provides naturalness, explanation facilities, exploitation of available cases and incremental learning capability.

Neuro-fuzzy modules could be used within CBR instead of plain neural or fuzzy components for tasks such as the ones described in Sections 3.2 and 3.3. For instance in [48] a distributed fuzzy neural network performs approximate matching to tolerate potential noise in retrieval, whereas in [114,128] the ANFIS neuro-fuzzy model and fuzzy ARTMAP are applied respectively to retrieve the most similar cases. Obviously non-embedded coupling approaches can also be applied.

3.7. Summary discussion for combinations of CBR with other intelligent methods

In this section, we summarize certain conclusions derived from the previous sections discussing combinations of CBR with other methods. Table 1 summarizes indicative uses of (single) other intelligent methods within CBR based on the discussion in Sections 3.1–3.5. Table 2 summarizes the ripeness of various coupling models in combinations of CBR with other intelligent methods based on the discussion in Sections 3.1–3.6. In Table 2, symbol '-' denotes minimal to null combination efforts, symbol '$\surd$-' denotes some number of combination efforts and symbol '$\surd$' denotes at least a fair number of combination efforts. A conclusion that can be derived from Table 2 is that there is room for extensive future research work involving non-embedded approaches combining CBR with other methods (with the exception of RBR). Table 3 summarizes complementary benefits derived when combining CBR with each one of the discussed intelligent methods.

Table 1
Indicative uses of other intelligent methods within CBR

Input problem (*or query*) *representation*	Ontologies (vocabulary, reduce amount of work required to input information, enforce domain model integrity, synonym problem handling, support inheritance of components, properties and relations from classes, creation of virtual components corresponding to complex structural relations and model verification)
Case representation	Fuzzy methods (fuzzy terms in cases, case base populated with fuzzy rules) Neural networks (analyze available data) Genetic algorithms (case feature discretization, irrelevant feature removal) Ontologies (vocabulary, structured representations, give semantic coherence and structure to cases)
Case abstraction	Ontologies
Initial case base construction	RBR Fuzzy methods (fuzzy similarity relations) Genetic algorithms
Case retrieval	RBR Fuzzy methods (multiple indexing of cases on a single feature with different degrees of membership, fuzzy similarity assessment and matching methods, fuzziness to assess temporal similarity, fuzzy analytical hierarchy process to retrieve relevant cases and then filter retrieved cases not complying to specific criteria, fuzzy rough sets assist in retrieving reusable cases, fuzzy clustering) Neural networks Genetic algorithms (assignment of feature weights, feature selection, selection of relevant indices for evolving environments, retrieval of multiple similar cases, finding the optimal k parameter if k nearest neighbor retrieval is applied, assist in organizing cases to clusters) Ontologies (indexing vocabulary, semantic similarity assessment, structural similarity)
Case adaptation	RBR (domain-independent and domain-specific adaptation rules) Fuzzy methods (fuzzy adaptation rules) Neural networks Genetic algorithms (produce more creative solutions and avoid production of some repetitive solutions) Ontologies (adaptation complies with relations and restrictions, representation of adaptation rules)
Case base maintenance	Fuzzy methods (fuzzy rules for case base reduction, fuzzy rules for generating cases in form of cluster granules reducing feature number and case base size, fuzzy rough sets for case feature reduction and weighting, fuzzy integrals determine case base competence) Neural networks Genetic algorithms (case base reduction, control case base quality and size, index selection in evolving environments, optimization of case base aspects)

Table 2
Ripeness of coupling models in combinations of CBR with other intelligent methods

	RBR	*Fuzzy Methods*	*Neural Networks*	*Genetic Algorithms*	*Ontologies*	*Neuro-Symbolic*	*Neuro-Fuzzy*
CBR embeds other method	√	√	√	√	√	–	√–
Other method embeds CBR	–	–	–	√	–	–	–
Sequential processing	√	√–	–	√–	–	–	–
Co-processing	√	√–	√–	√–	–	√–	–

4. Conclusions

In this paper, we discuss key aspects involving combinations of CBR with other intelligent methods. Such combinations are becoming increasingly popular due to the fact that in many application domains a vast amount of case data is available. Such combined approaches have managed to solve problems in application domains where a case-based module needs the assistance and/or completion of other intelligent modules in order to produce effective results. This trend is very likely to carry on in the following years.

The discussion concerning combinations of CBR should cover various aspects of these combinations such as, degree of coupling between the combined components, information flow, reasoning mode and reasoning control. Furthermore, it should take into consideration Medsker's categorization scheme which, as already mentioned, although is generally endorsed by hybrid intelligent systems community, it has not been paid attention to by the community working towards combinations of CBR with other methods.

Future directions in combinations of CBR with other intelligent methods could involve a number of aspects.

Table 3
Benefits derived by combining CBR with each other intelligent method

CBR with RBR	RBR: rule-based domain knowledge, general knowledge, naturalness, compact knowledge representation, rule-based inference and explanation facilities, classification capabilities. CBR: easier knowledge acquisition, empirical knowledge, naturalness, reasoning not performed from scratch, reasoning handles unexpected/missing inputs, generalization, learning capabilities, adaptability, case-based explanation facilities.
CBR with fuzzy logic	*Fuzzy logic:* imprecision handling, domain knowledge, the compact knowledge representation of fuzzy rules, explanation facilities of fuzzy RBR. CBR: empirical knowledge, learning capabilities, reasoning not performed from scratch as in fuzzy RBR.
CBR with neural networks	*Neural networks:* efficiency, robustness, exploitation of available training examples, learning capabilities, generalization, classification/clustering capabilities CBR: naturalness, modularity, empirical knowledge, incremental learning, explanation facilities, transfer of knowledge to other related application domains
CBR with genetic algorithms	*Genetic algorithms:* problem solution searching, optimization, adaptability, compact representation of problem parameters as numeric values and representation of a variable number of (not predetermined) possible solutions. CBR: experience, naturalness, learning capabilities, explanation facilities
CBR with ontologies	*Ontologies:* information-rich structure, vocabulary, formalize explicitly declared relevant knowledge, facilitate knowledge sharing and reuse, assist in handling, integrating and reasoning from distributed data and information on the Web, Semantic Web technologies, assist in implementation of knowledge engineering tools aiming to reuse, flexibility and usability. CBR: empirical knowledge, naturalness, reasoning capabilities, learning capabilities, explanation facilities
CBR with neuro-symbolic methods	*Neuro-symbolic methods:* efficiency, robustness, exploitation of available training examples and symbolic domain knowledge, learning capabilities, generalization, classification/clustering capabilities CBR: empirical knowledge, naturalness, modularity, incremental learning, explanation facilities
CBR with neuro-fuzzy methods	*Neuro-fuzzy methods:* imprecision handling, efficiency, robustness, exploitation of available training examples and fuzzy domain knowledge, learning capabilities, generalization, classification/clustering capabilities CBR: empirical knowledge, naturalness, modularity, incremental learning, explanation facilities

Main such aspects involve: (a) combinations of CBR with soft computing methods, (b) combinations of CBR with fuzzy rules, (c) combinations of CBR with ontologies and (d) combinations of CBR with neuro-symbolic and neuro-fuzzy approaches.

Combinations of CBR with soft computing methods not following an embedded coupling approach could be an interesting future research direction. At present there seems to be a lack of great interest in pursuing this direction since the main interest has focused on employing soft computing methods within CBR. A non-embedded direction in the combinations of CBR with soft computing could be pursued as thoroughly as in the case of combinations of CBR with RBR/MBR. A further step towards this direction could involve non-embedded multi-integrated approaches combining CBR with multiple soft computing methods or combinations of CBR, soft computing and other intelligent methods (e.g. RBR, MBR or ontologies).

Combinations of CBR with fuzzy rule-based systems could be based on work combining CBR with RBR that is, investigation of various coupling approaches.

The increasing interest in Web-based intelligent systems and future advances in the Semantic Web is likely to provide an impetus to approaches combining CBR with ontologies. This trend is likely to involve multi-integrated approaches combining CBR, ontologies and other intelligent methods.

Finally, a direction that may be useful to be pursued involves non-embedded coupling approaches combining CBR with neuro-symbolic and neuro-fuzzy modules. Few such approaches have been developed.

A general conclusion that can be derived from the study of combinations of CBR with other methods is that a number of directions are open to extensive future research especially in non-embedded combined approaches. It is likely that such approaches will become more and more mature and established. In such a case, critical necessities will involve availability of tools to build combined systems and exploit practical experience of applying combined approaches to several real-world applications. It would be worthwhile for such tools to support several types of combined approaches from simper to more sophisticated ones. This would assist in fast implementation of systems following combined approaches.

References

[1] A. Aamodt and E. Plaza, Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches, *AI Communications* **7** (1994), 39–59.

[2] A. Abraham, Intelligent Systems: Architectures and Perspectives, in: *Recent Advances in Intelligent Paradigms and Applications*, *Studies in Fuzziness and Soft Computing*, J. Kacprzyk, A. Abraham and L.C. Jain, eds, Physica-Verlag, 2003, pp. 1–35.

[3] G. Agre, A Two-Tiered Reasoning and Learning Architecture, in: *Proceedings of the 1998 AAAI Spring Symposium on Multimodal Reasoning*, AAAI Technical Report SS-98-04, AAAI Press, 1998, pp. 124–129.

[4] G. Agre and I. Koprinska, Case-Based Refinement of Knowledge-Based Neural Networks, in: *Proceedings of the International Conference "Intelligent Systems: A Semiotic Perspective"*, J. Albus, A. Meystel and R. Quintero, eds, Volume II: Applied Semiotics, US Government Printing Office, Gaithersburg, MD, USA, 1996, pp. 221–226.

[5] H. Ahn and K. Kim, Global Optimization of Case-Based Reasoning for Breast Cytology Diagnosis, *Expert Systems with Applications* **36** (2009), 724–734.

[6] G. Allampalli-Nagaraj and I. Bichindaritz, Automatic Semantic Indexing of Medical Images using a Web Ontology Language for Case-Based Image Retrieval, *Engineering Applications of Artificial Intelligence* **22** (2009), 18–25.

[7] S. Bader and P. Hitzler, Dimensions of Neural-Symbolic Integration – a Structured Survey, in: *We Will Show Them: Essays in Honour of Dov Gabbay*, (Vol. 1), S. Artemov, H. Barringer, A.S. d'Avila Garcez, L.C. Lamb and J. Woods, eds, International Federation for Computational Logic, College Publications, 2005, pp. 167–194.

[8] C. Balducelli and C. D' Esposito, Genetic Agents in an EDSS System to Optimize Resources Management and Risk Object Evacuation, *Safety Science* **35** (2000), 59–73.

[9] I. Bischindaritz, Memoire: A Framework for Semantic Interoperability of Case-Based Reasoning Systems in Biology and Medicine, *Artificial Intelligence in Medicine* **36** (2006), 177–192.

[10] I. Bichindaritz, Case-Based Reasoning in the Health Sciences, *Artificial Intelligence in Medicine* **36** (2006), 121–125.

[11] I. Bichindaritz and K.M. Sullivan, Generating Practice Cases for Medical Training from a Knowledge-Based Decision Support System, in: *Proceedings of the Workshop on Case-based Reasoning for Education and Training, Sixth European Conference on Case-based Reasoning*, Robert Gordon University, Aberdeen, Scotland, UK, 2002, pp. 3–14.

[12] P.P. Bonissone and R. Lopez de Mantaras, Fuzzy Case-Based Reasoning Systems, *Handbook on Fuzzy Computing*, (Vol. F 4.3), Oxford University Press, 1998.

[13] L. Castillo, E. Armengol, E. Onaindia, L. Sebastia, J. Gonzalez-Boticario, A. Rodriguez, S. Fernandez, J.D. Arias and D. Borrajo, SAMAP: An User Oriented Adaptive System for Planning Tourist Visits, *Expert Systems with Applications* **34** (2008), 1318–1332.

[14] L. Ceccaroni, U. Cortes and M. Sanchez-Marre, OntoWEDSS: Augmenting Environmental Decision-Support Systems with Ontologies, *Environmental Modelling & Software* **19** (2004), 785–797.

[15] A.L. Cechin and E. Battistella, The Interpretation of Feedforward Neural Networks for Secondary Structure Prediction using Sugeno Fuzzy Rules, *International Journal of Hybrid Intelligent Systems* **4** (2007), 3–16.

[16] P.-C. Chang, J.-C. Hsieh and C.-H. Liu, A Case-Injected Genetic Algorithm for Single Machine Scheduling Problems with Release Time, *International Journal on Production Economics* **103** (2006), 551–564.

[17] P.-C. Chang and C.-Y. Lai, A Hybrid System Combining Self-Organizing Maps with Case-Based Reasoning in Wholesaler's New-Release Book Forecasting, *Expert Systems with Applications* **29** (2005), 183–192.

[18] P.-C. Chang, C.-Y. Lai and K.R. Lai, A Hybrid System by Evolving Case-Based Reasoning with Genetic Algorithm in Wholesaler's Returning Book Forecasting, *Decision Support Systems* **42** (2006), 1715–1729.

[19] P.-C. Chang, C.-H. Liu and R.K. Lai, A Fuzzy Case-Based Reasoning Model for Sales Forecasting in Print Circuit Board Industries, *Expert Systems with Applications* **34** (2008), 2049–2058.

[20] W. Cheetam, S.C.K. Shiu and R.O. Weber, Soft Case-Based Reasoning, *Knowledge Engineering Review* **20** (2006), 267–269.

[21] D. Chen and P. Burrell, Case-Based Reasoning System and Artificial Neural Networks: A Review, *Neural Computing & Applications* **10** (2001), 264–276.

[22] J.-H. Chen and S.C. Hsu, Hybrid ANN-CBR Model for Disputed Change Orders in Construction Projects, *Automation in Construction* **17** (2007), 56–64.

[23] R.M. Colomb and M.N. Ahmad, Merging ontologies requires interlocking institutional worlds, *Applied Ontology* **2** (2007), 1–12.

[24] Common Logic Working Group Documents: http://common-logic.org/.

[25] S. Craw, J. Jarmulak and R. Rowe, Maintaining Retrieval Knowledge in a Case-Base Reasoning System, Computational Intelligence **17** (2001), 346–363.

[26] DAML Web Page: http://www.daml.org/about.html.

[27] DAML+OIL Web Page: http://www.daml.org/2001/03/daml+oil-index.

[28] R.L. De Mantaras, D. McSherry, D. Bridge, D. Leake, B. Smyth, S. Craw, B. Faltings, M.L. Maher, M.T. Cox, K. Forbus, M. Keane, A. Aamodt and I. Watson, Retrieval, Reuse, Revision, and Retention in Case-Based Reasoning, *The Knowledge Engineering Review* **20** (2005), 215–240.

[29] B. Diaz-Agudo, P.A. Gonzalez-Calero and J.A. Recio-Garcia, Building CBR Systems with jCOLIBRI, *Science of Computer Programming* **69** (2007), 68–75.

[30] S. Dutta and P.P. Bonissone, Integrating Case Based and Rule Based Reasoning, *International Journal of Approximate Reasoning* **8** (1993), 163–203.

[31] R.-J. Dzeng and H.-Y. Lee, Critiquing Contractors' Scheduling by Integrating Rule-Based and Case-Based Reasoning, *Automation in Construction* **13** (2004), 665–678.

[32] K. Evans-Romaine and C. Marling, Prescribing Exercise Regimens for Cardiac and Pulmonary Disease Patients with CBR, in: *Proceedings of the Workshop on CBR in the Health Services, Fifth International Conference on Case-based Reasoning*, Trondheim, Norway, 2003, pp. 45–52.

[33] F. Fdez-Riverola and J.M. Corchado, FSfRT: Forecasting System for Red Tides, *Applied* Intelligence **21** (2004), 251–264.

[34] L.M. Fu, Knowledge-Based Connectionism for Revising Domain Theories, *IEEE Transactions on Systems, Man and Cybernetics* **23** (1993), 173–182.

[35] Y. Fu and R. Shen, GA Based CBR Approach in Q&A system, *Expert Systems with Applications* **26** (2004), 167–170.

[36] J.M. Garrell i Guiu, E. Golobardes i Ribé, E. Bernadoí Mansilla and X. Lloraì Fabrega, Automatic Diagnosis with Genetic Algorithms and Case-Based Reasoning, *Artificial Intelligence in Engineering* **13** (1999), 367–372.

[37] V.C. Georgopoulos and C.D. Stylios, Complementary Case-Based Reasoning and Competitive Fuzzy Cognitive Maps for Advanced Medical Decisions, *Soft Computing* **12** (2008), 191–199.

[38] P. Gervas, B. Diaz-Agudo, F. Peinado and R. Hervas, Story Plot Generation Based on CBR, *Knowledge-Based Systems* **18** (2005), 235–242.

[39] A.R. Golding and P.S. Rosenbloom, Improving Accuracy by Combining Rule-Based and Case-Based Reasoning, *Artificial Intelligence* **87** (1996), 215–254.

[40] A. Hanemann and P. Marcu, Algorithm Design and Application of Service-Oriented Event Correlation, in: *Proceedings of the 3rd IEEE/IFIP International Workshop on Business-driven IT Management*, IEEE, 2008, pp. 61–70.

[41] I. Hatzilygeroudis and J. Prentzas, Integrating (Rules, Neural Networks) and Cases for Knowledge Representation and Reasoning in Expert Systems, *Expert Systems with Applications* **27** (2004), 63–75.

[42] I. Hatzilygeroudis and J. Prentzas, Neuro-Symbolic Approaches for Knowledge Representation in Expert Systems, *International Journal of Hybrid Intelligent Systems* **1** (2004), 111–126.

[43] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall, 1999.

[44] M.-J. Huang, H.-S. Huang and M.-Y. Chen, Constructing a Personalized E-learning System Based on Genetic Algorithm and Case-Based Reasoning Approach, *Expert Systems with Applications* **33** (2007), 551–564.

[45] B. W. Huang, M. L. Shih, N.-H. Chiu, W. Y. Hu and C. Chiu, Price Information Evaluation and Prediction for Broiler using Adapted Case-Based Reasoning Approach, *Expert Systems with Applications* **36** (2009), 1014–1019.

[46] S.C. Hui, A.C.M. Fong and G. Jha, A Web-Based Intelligent Fault Diagnosis System for Customer Service Support, *Engineering Applications of Artificial Intelligence* **14** (2001), 537–548.

[47] C. Hsu, C. Chiu and P.-L. Hsu, Predicting Information Systems Outsourcing Success Using a Hierarchical Design of Case-Based Reasoning, *Expert Systems with Applications* **26** (2004), 435–441.

[48] C.-C. Hsu and C.-S. Ho, A New Hybrid Case-Based Architecture for Medical Diagnosis, *Information Sciences* **166** (2004), 231–247.

[49] K.H. Im and S.C. Park, Case-Based Reasoning and Neural Network Based Expert System for Personalization, *Expert Systems with Applications* **32** (2007), 77–85.

[50] V.H. Jacobo, A. Ortiz, Y. Cerrud and R. Schouwenaars, Hybrid Expert System for the Failure Analysis of Mechanical Elements, *Engineering Failure Analysis* **14** (2007), 1435–1443.

[51] Y.-J. Jiang, J. Chen and X.-Y. Ruan, Fuzzy Similarity-Based Rough Set Method for Case-Based Reasoning and its Application in Tool Selection, *International Journal of Machine Tools & Manufacture* **46** (2006), 107–113.

[52] Y.-K. Juan, S.-G. Shih and Y.-H. Perng, Decision Support for Housing Customization: A Hybrid Approach using Case-Based Reasoning and Genetic Algorithm, *Expert Systems with Applications* **31** (2006), 83–93.

[53] J.M. Juarez, F. Guil, J. Palma and R. Marin, Temporal Similarity by Measuring Possibilistic Uncertainty in CBR, *Fuzzy Sets and Systems* **160** (2009), 214–230.

[54] A. Khanum, M. Mufti, M.Y. Javed and M.Z. Shafiq, Fuzzy Case-Based Reasoning for Facial Expression Recognition, *Fuzzy Sets and Systems* **160** (2009), 231–250.

[55] K.-J. Kim, Toward Global Optimization of Case-Based Reasoning Systems for Financial Forecasting, *Applied Intelligence* **21** (2004), 239–249.

[56] G.-H. Kim, S.-H. An and K.-I. Kang, Comparison of Construction Cost Estimating Models Based on Regression Analysis, Neural Networks, and Case-Based Reasoning, *Building and Environment* **39** (2004), 1235–1242.

[57] K.-J. Kim and H. Ahn, A Recommender System using GA K-means Clustering in an Online Shopping Market, *Expert Systems with Applications* **34** (2008), 1200–1209.

[58] Knowledge Interchange Format: http://www.ksl.stanford.edu/knowledge-sharing/kif/.

[59] J.L. Kolodner, *Case-Based Reasoning*, Morgan Kaufmann, 1993.

[60] R.J. Kuo, Y.P. Kuo and K.-Y. Chen, Developing a Diagnostic System through Integration of Fuzzy Case-Based Reasoning and Fuzzy Ant Colony System, *Expert Systems with Applications* **28** (2005), 783–797.

[61] O.B. Kwon, Meta Web Service: Building Web-based Open Decision Support System based on Web Services, *Expert Systems with Applications* **24** (2003), 375–389.

[62] O. Kwon and M. Kim, MyMessage: Case-Based Reasoning and Multicriteria Decision Making Techniques for Intelligent Context-Aware Message Filtering, *Expert Systems with Applications* **27** (2004), 467–480.

[63] A. Lau, E. Tsui and W.B. Lee, An Ontology-Based Similarity Measurement for Problem-Based Case Reasoning, *Expert Systems with Applications* **36** (2009), 6574–6579.

[64] D.B. Leake (Ed.), *Case-Based Reasoning: Experiences, Lessons, and Future Directions*, AAAI Press, 1996.

[65] G. H. Lee, Rule-Based and Case-Based Reasoning Approach for Internal Audit of Bank, *Knowledge-Based Systems* **21** (2008), 140–147.

[66] M.R. Lee, An Exception Handling of Rule-Based Reasoning using Case-Based Reasoning, *Journal of Intelligent and Robotic Systems* **35** (2002), 327–338.

[67] S.-T. Li and H.-F. Ho, Predicting Financial Activity with Evolutionary Fuzzy Case-Based Reasoning, *Expert Systems with Applications* **36** (2009), 411–422.

[68] T.W. Liao, An Investigation of a Hybrid CBR Method for Failure Mechanisms Identification, *Engineering Applications of Artificial Intelligence* **17** (2004), 123–134.

[69] K.F.-R. Liu and C.-W. Yu, Integrating Case-Based and Fuzzy Reasoning to Qualitatively Predict Risk in an Environmental Impact Assessment Review, *Environmental Modelling & Software* (to appear).

[70] C.G. Looney and L.R. Liang, Cognitive Situation and Threat Assessments of Ground Battlespaces, *Information Fusion* **4** (2003), 297–308.

[71] Z. Lou, H. Jiang and X. Ruan, Development of an Integrated Knowledge-Based System for Mold-Base Design, *Journal of Materials Processing Technology* **150** (2004), 194–199.

[72] S.J. Louis, G. McGraw and R.O. Wyckoff, Case-Based Reasoning Assisted Explanation of Genetic Algorithm Results, *Journal of Experimental & Theoretical Artificial Intelligence* **5** (1993), 21–37.

[73] S. J. Louis and C. Miles, Playing to Learn: Case-Injected Genetic Algorithms for Learning to Play Computer Games, *IEEE Transactions on Evolutionary Computation* **9** (2005), 669–681.

[74] M. Malek and A. Labbi, *Integration of Case-Based Reasoning and Neural Networks Approaches for Classification*, Technical Report, TIMC IMAG, LIFIA IMAG, Grenoble, France, 1995.

[75] C.R. Marling, G.J. Petot and L.S. Sterling, Integrating Case-Based and Rule-Based Reasoning to Meet Multiple Design Constraints, *Computational Intelligence* **15** (1999), 308–332.
[76] C. Marling, E. Rissland and A. Aamodt, Integrations with Case-Based Reasoning, *Knowledge Engineering Review* **20** (2006), 241–245.
[77] C. Marling, M. Sqalli, E. Rissland, H. Munoz-Avila and D. Aha, Case-Based Reasoning Integrations, *AI Magazine* **23** (2002), 69–86.
[78] L.R. Medsker, *Hybrid Intelligent Systems*, Kluwer Academic Publishers, Second Printing, 1998.
[79] G. B. Melton, S. Parsons, F. P. Morrison, A. S. Rothschild, M. Markatou and G. Hripcsak, Inter-patient Distance Metrics using SNOMED CT Defining Relationships, *Journal of Biomedical Informatics* **39** (2006), 697–705.
[80] R. Mitra and J. Basak, Methods of Case Adaptation: a Survey, *International Journal of Intelligent Systems* **20** (2005), 627–645.
[81] S. Montani and R. Bellazzi, Supporting Decisions in Medical Applications: the Knowledge Management Perspective, *International Journal of Medical Informatics* **68** (2002), 79–90.
[82] S. Montani, P. Magni, R. Bellazzi, C. Larizza, A.V. Roudsari and E.R. Carson, Integrating Model-Based Decision Support in a Multi-Modal Reasoning System for Managing Type 1 Diabetic Patients, *Artificial Intelligence in Medicine* **29** (2003), 131–151.
[83] D. Nauck, F. Klawonn and R. Kruse, *Foundations of Neuro-Fuzzy Systems*, John Wiley, 1997.
[84] OIL Web Page: http://www.ontoknowledge.org/oil/.
[85] D. Ourston and R.J. Mooney, Theory Refinement Combining Analytical and Empirical Methods, *Artificial Intelligence* **66** (1994), 273–310.
[86] OWL Working Group: http://www.w3.org/2007/OWL/wiki/OWL_Working_Group.
[87] S.K. Pal and P. Mitra, Case Generation Using Rough Sets with Fuzzy Discretization, *IEEE Transactions on Knowledge and Data Engineering* **16** (2004), 292–300.
[88] S. K. Pal and S.C.K. Shiu, *Foundations of Soft Case-Based Reasoning*, John Wiley, 2004.
[89] B. Pandey and R.B. Mishra, An Integrated Intelligent Computing Model for the Interpretation of EMG Based Neuromuscular Diseases, *Expert Systems with Applications* **36** (2009), 9201–9213.
[90] H.-J. Park, J.-S. Oh, D.-U. Jeong and K.-S. Park, Automated Sleep Stage Scoring using Hybrid Rule- and Case-Based Reasoning, *Computers and Biomedical Research* **33** (2000), 330–349.
[91] S. Passone, P.W.H. Chung and V. Nassehi, Incorporating Domain-Specific Knowledge into a Genetic Algorithm to Implement Case-Based Reasoning Adaptation, *Knowledge-Based Systems* **19** (2006), 192–201.
[92] R. Pavon, F. Dνaz, R. Laza and V. Luzon, Automatic Parameter Tuning with a Bayesian Case-Based Reasoning System. A Case of Study, *Expert Systems with Applications* **36** (2009), 3407–3420.
[93] E.I. Perez, C.A. Coello Coello and A. Hernandez-Aguirre, Extraction and Reuse of Design Patterns from Genetic Algorithms using Case-Based Reasoning, *Soft Computing* **9** (2005), 44–53.
[94] N.H. Phuong, N.R. Prasad, D.H. Hung and J.T. Drake, Approach to Combining Case Based Reasoning with Rule Based Reasoning for Lung Disease Diagnosis, Proceedings of the Joint Ninth International Fuzzy Systems Congress and Twentieth International Conference of the North American Fuzzy Information Processing Association World Society, *IEEE* **2** (2001), 883–888.
[95] C.A. Policastro, A.C.P.L.F. Carvalho and A.C.B. Delbem, Hybrid Approaches for Case Retrieval and Adaptation, in: KI 2003: Advances in Artificial Intelligence, *Lecture Notes in Computer Science* **2821** (2003), 297–311.
[96] J. Prentzas and I. Hatzilygeroudis, Categorizing Approaches Combining Rule-Based and Case-Based Reasoning, *Expert Systems* **24** (2007), 97–122.
[97] J. Prentzas and I. Hatzilygeroudis, Combinations of Case-Based Reasoning with Other Intelligent Methods, in: *Proceedings of the 1st International Workshop on Combinations of Intelligent Methods and Applications* (*CIMA 2008*) *in conjunction with 18th European Conference on Artificial Intelligence* (*ECAI-08*), I. Hatzilygeroudis, C. Koutsojannis and V. Palade, eds, Patras, Greece, 2008, pp. 55–58.
[98] J. Prentzas, I. Hatzilygeroudis and O. Michail, Improving the Integration of Neuro-Symbolic Rules with Case-Based Reasoning, in: *Artificial Intelligence: Theories, Models and Applications*, *Lecture Notes in Artificial Intelligence*, (Vol. 5138), J. Darzentas, G.A. Vouros, S. Vosinakis and A. Arnellos, eds, Springer-Verlag, 2008, pp. 377–382.
[99] P. Pulkkinen, M. Laurikkala, A. Ropponen and H. Koivisto, Quality Management in GPRS Networks with Fuzzy Case-Based Reasoning, *Knowledge-Based Systems* **21** (2008), 421–428.
[100] E. Ramos, H. Nunez and R. Casanas, A Fuzzy Hybrid Intelligent System for Human Semen Analysis, in: *Advances in Artificial Intelligence – IBERAMIA 2008*, *Lecture Notes in Artificial Intelligence*, (Vol. 5290), H. Geffner, R. Prada, I. Machado Alexandre and N. David, eds, Springer-Verlag, 2008, pp. 422–431.
[101] D.V. Rao and V.V.S. Sarma, A Rough – Fuzzy Approach for Retrieval of Candidate Components for Software Reuse, *Pattern Recognition Letters* **24** (2003), 875–886.
[102] RIF Working Group: http://www.w3.org/2005/rules/wiki/RIF_Working_Group.
[103] E.L. Rissland and D.B. Skalak, CABARET: Rule Interpretation in a Hybrid Architecture, *International Journal of Man-Machine Studies* **34** (1991), 839–887.
[104] Y. Rodriguez, C. Morell, R. Grau, M.M. Garcia and B. De Baets, Fuzzy Case-Based System for Classification Tasks on Missing and Noisy Data, in: *Proceedings of the 8th International Conference on Hybrid Intelligent Systems*, IEEE Computer Society, 2008, pp. 222–227.
[105] D. Rossille, J.-F. Laurent and A. Burgun, Modeling a Decision Support System for Oncology using Rule-Based and Case-Based Reasoning Methodologies, *International Journal of Medical Informatics* **74** (2005), 299–306.
[106] K.M. Saridakis and A.J. Dentsoras, Case-DeSC: A System for Case-Based Design with Soft Computing Techniques, *Expert Systems with Applications* **32** (2007), 641–657.
[107] R. Schmidt and O. Vorobieva, Case-Based Reasoning to Explain Medical Model Exceptions, in: *eHealth Beyond the Horizon – Get IT There – Proceedings of the XXIst International Congress of the European Federation for Medical Informatics, Studies in Health Technology and Informatics*, (Volume 136), S.K. Andersen, G.O. Klein, S. Schulz, J. Aarts and M.C. Mazzoleni, eds, IOS Press, 2008, pp. 259–264.
[108] S.C.K. Shiu, Y. Li and X.Z. Wang, Using Fuzzy Integral to Model Case-Base Competence, in: *Proceedings of the Workshop on Soft Computing in Case-Based Reasoning, 4th Inter-*

national Conference in Case-Based Reasoning, Vancouver, British Columbia, Canada, 2001, pp. 206–212.

[109] S.C.K. Shiu and S.K. Pal, Case-Based Reasoning: Concepts, Features and Soft Computing, *Applied Intelligence* **21** (2004), 233–238.

[110] S.C.K. Shiu, D.S. Yeung, C.H. Sun and X.Z. Wang, Transferring Case Knowledge To Adaptation Knowledge: An Approach for Case-Base Maintenance, *Computational Intelligence* **17** (2001), 295–314.

[111] L.K. Soh and C. Tsatsoulis, Combining Genetic Algorithms and Case-Based Reasoning for Genetic Learning of a Case-base: A Conceptual Framework, in: *Proceedings of the Genetic and Evolutionary Computation Conference*, Morgan Kaufmann Publishers, 2001, pp. 376–383.

[112] F. Sormo, J. Cassens and A. Aamodt, Explanation in Case-Based Reasoning – Perspectives and Goals, *Artificial Intelligence Review* **24** (2005), 109–143.

[113] Z. Sun, G. Finnie and K. Weber, Case Base Building with Similarity Relations, *Information Sciences* **165** (2004), 21–43.

[114] K.H. Tan, C.P. Lim, K. Platts and H.S. Koay, An Intelligent Decision Support System for Manufacturing Technology Investments, *International Journal of Production Economics* **104** (2006), 179–190.

[115] A.B. Tickle, R. Andrews, M. Golea and J. Diederich, The Truth will Come to Light: Directions and Challenges in Extracting the Knowledge Embedded within Trained Artificial Neural Networks, *IEEE Transactions on Neural Networks* **9** (1998), 1057–1068.

[116] G. Towell and J. Shavlik, Knowledge-Based Artificial Neural Networks, *Artificial Intelligence* **70** (1994), 119–165.

[117] W.M. Wang, C.F. Cheung, W.B. Lee and S.K. Kwok, Knowledge-Based Treatment Planning for Adolescent Early Intervention of Mental Healthcare: a Hybrid Case-Based Reasoning Approach, *Expert Systems* **24** (2007), 232–251.

[118] I. Watson, Case-Based Reasoning is a Methodology not a Technology, *Knowledge-Based Systems* **12** (1997), 303–308.

[119] I. Watson and F. Marir, Case-Based Reasoning: A Review, *The Knowledge Engineering Review* **9** (1994), 355–381.

[120] R. Weber, Fuzzy Set Theory and Uncertainty in Case-Based Reasoning, *Engineering Intelligent Systems for Electrical Engineering and Communications* **14** (2006), 121–136.

[121] D. Wilson and D. Leake, Maintaining Case-Based Reasoners: Dimensions and Directions, *Computational Intelligence* **17** (2001), 196–213.

[122] P. Wriggers, M. Siplivaya, I. Joukova and R. Slivin, Intelligent Support of Engineering Analysis using Ontology and Case-Based Reasoning, *Engineering Applications of Artificial Intelligence* **20** (2007), 709–720.

[123] M.-C. Wu, Y.-F. Lo and S.-H. Hsu, A Fuzzy CBR Technique for Generating Product Ideas, *Expert Systems with Applications* **34** (2008), 530–540.

[124] B. Wyns, L. Boullart, S. Sette, D. Baeten, I. Hoffman and F. De Keyser, Prediction of Arthritis using a Modified Kohonen Mapping and Case Based Reasoning, *Engineering Applications of Artificial Intelligence* **17** (2004), 205–211.

[125] B. Wyns, L. Boullart, S. Sette, D. Baeten, I. Hoffman and F. De Keyser, Prediction of Arthritis using a Modified Kohonen Mapping and Case Based Reasoning, *Engineering Applications of Artificial Intelligence* **17** (2004), 205–211.

[126] B.-S. Yang, T. Han and Y.-S. Kim, Integration of ART-Kohonen Neural Network and Case-Based Reasoning for Intelligent Fault Diagnosis, *Expert Systems with Applications* **26** (2004), 387–395.

[127] H.-L. Yang and C.-S. Wang, Two Stages of Case-Based Reasoning – Integrating Genetic Algorithm with Data Mining Mechanism, *Expert Systems with Applications* **35** (2008), 262–272.

[128] W.-D. Yu and Y.-C. Liu, Hybridization of CBR and Numeric Soft Computing Techniques for Mining of Scarce Construction Databases, *Automation in Construction* **15** (2006), 33–46.

[129] F.-C. Yuan and C. Chiu, A Hierarchical Design of Case-Based Reasoning in the Balanced Scorecard Application, *Expert Systems with Applications* **36** (2009), 333–342.

[130] J. Zhao, L. Cui, L. Zhao, T. Qiu and B. Chen, Learning HAZOP Expert System by Case-Based Reasoning and Ontology, *Computers and Chemical Engineering* **33** (2009), 371–378.

[131] Z.Y. Zhuang, L. Churilov, F. Burstein and K. Sikaris, Combining Data Mining and Case-Based Reasoning for Intelligent Decision Support for Pathology Ordering by General Practitioners, *European Journal of Operational Research* **195** (2009), 662–675.