Cases on fire: applying CBR to emergency management

Francesco Ricci, Paolo Avesani *and* **Anna Perini**
Istituto per la Ricerca Scientifica e Tecnologica, via Sommarive, 38050 Povo (TN), Italy
(e-mail: *ricci@itc.it*), e-mail:*avesani@itc.it*) *and* (e-mail: *perini@itc.it*)

This paper illustrates two applications of the case-based reasoning methodology (CBR) to the forest fire emergencies application domain. The first application is an interactive planning system used for planning the initial attack on forest fires. Our planning approach is built on the integration of case-based reasoning and constraint reasoning. Case-based reasoning plays the role of assumption maker, suggesting old solution to similar situation, and constraint reasoning filters those assumption in a feasible solution. The second application combines case-based reasoning and knowledge-discovering techniques to help both teachers and students to acquire the knowledge contained in a case base of past interventions. Manipulation, browsing and enquiring tools enable users to directly extract knowledge in the form of: feature mutual dependencies; clusters and prototypes existing in the case base; feature statistical descriptors; results of CBR queries.

1. Introduction

This paper illustrates two applications of Case-Based Reasoning techniques to the forest fire emergencies application domain. The first is an interactive planning system devoted to plan the initial attack to forest fires. The system is part of an intelligent decision support system aimed at supporting the user in the whole process of fire fighting including both situation assessment and planning activities.[1]

Our planning approach is built on the integration of case-based reasoning[2] and constraint reasoning.[3] The case based module retrieves partial plans from memory using a novel similarity metric that has a local definition and can be adapted by a reinforcement learning procedure.[4] Plans are represented in a hierarchy of parts and partial plans can be completed using the similarity metric by searching for expansions of the leaves in the plan tree structure.

The constraint module adapts the retrieved plan, attaching and propagating constraints. Constraint reasoning is used mainly for managing temporal relations defined on sub-plans or between actions in plans. Constraints are also used in the identification of the sets of applicable actions and suitable resources in relation to particular scenario data. The temporal dimension of the plan is managed with a quantitative approach based on temporal variables over continuous domains. The constraint module supports also the resource allocation process for the chosen plan.

The second application combines case-based reasoning and knowledge-discovering techniques to help both teachers and students to acquire the knowledge contained in a case base. In our application, cases record the forest fire emergencies which have occurred in a department of the southern France. Manipulation, browsing and enquiring tools enable users to directly extract knowledge in the form of: feature mutual dependencies; clusters and prototypes existing in the case base; feature statistical descriptors; results of CBR queries. In fact, the major focus of the system is retrieval for learning, that is addressed by using both CBR and KDD (Knowledge discovery in data bases) techniques.[5] So, for example, retrieval based on a nearest neighbor algorithm, a standard techniques in CBR,[6] can be supported by a selection of relevant features, performed with statistical and information theory algorithms.[7] CBET supports the user from the definition of the data structures, to the modification and maintenance of the case base.

The paper is organized as follows. First we recall the CBR methodology, its advantages with respect to more traditional expert systems techniques and its open problems. Then we briefly collect some information on the application domain and the application problems that have been addressed. The following sections describe the planning application and the data mining application of CBR to emergency management. We conclude by quoting a few similar approaches.

2. Case-based reasoning

Case-Based Reasoning (CBR) is a very general problem solving methodology with a simple and appealing definition: when faced with a new problem, search the memory for a similar one and reuse its solution to the current problem.[2] Humans seem to rely on that approach for a number of mundane problems like: finding their way to get home; filling up the tax forms; deciding the menu for the Christmas lunch, etc. When the memory fails, humans combine recall with explicit solution of the unsolved subgoals.

Expert Systems based on the CBR methodology are becoming more and more popular. The applications range from help desk[8,9] to planning,[10] from design[11] to legal reasoning.[12] A number of tools[13, 14] is feeding a growing market which is replacing the classical market of expert system shells.

There are two main reasons for such a popularity. First, a fast growing number of case bases are becoming available as side effect of the introduction of Information Technology in the organizations. Second, compared with a more traditional rule based system, a CBR system seems to speed up the knowledge engineering process. The costly and difficult task of rule extraction is replaced by the apparently simpler activity of case definition and case base compilation.

In fact many features of the CBR methodology are critical and may become technical risks to carefully take under control in an application. First of all, all CBR systems are based on a distance metric[7] that evaluates the similarity between cases. That is crucial in the first step of the CBR solving process, namely the retrieval from the memory. Cases are normally described by etherogeneous features whose contributions to the distance computations have to be integrated in a reasonable way.[15] Moreover, when cases are stored using more structured representations like labeled graphs there is not a definitive solution to the similarity assessment and case retrieval yet.[11] Second, the reuse of a stored case often requires an adaptation step where dissimilarities between the retrieved case and the problem case have to be considered when building the solution.[2] That again has not a straightforward solution and it might be the case that adapting a solution could be even more difficult than building a new one from scratch.[16]

In the following sections we shall illustrate how these problems have been addressed in our applications to forest fire management.

3. Emergency management: the forest fire domain

The natural emergencies domain is an application field that sets stunning problems for IT. When a disaster like a forest fire starts a complex workflow is activated. An immediate assessment of the situation is evaluated in the "crisis room". The exact dimension of the fire is acquired using remote sensing or by a direct connection with a scout at the scene of the fire. The meteo conditions, the state of the resources (humans and means), the accessibility of the fire zone, the type of vegetation, the presence of nearby civil structures, are all considered to make a clear picture of the situation. IT tools like Geographical Information Systems (GIS), distributed data bases, mathematical fire spreading models are now in normal use and contribute to tackle the "intelligence" phase of what is becoming a very complex decision making process.[17]

When assessment is satisfactorily completed the alternatives should be considered. The fire can be fought along the flanks or right from the front. Civilian could be rescued sending helicopters or simply suggesting them a path to follow. The fire can be contained with retardant bombing and then blocked with a fire line or a certain number of fire squads must be sent to spray water taking advantage of a country road that pass close to the left flank of the fire. These are only a few examples of actions that a decision maker could list among alternative elementary components of a more complex and integrated course of actions. We describe this stage as a planning problem:[18] the state of the environment, which was acquired in the "intelligence" phase, and a collection of attainable goals constitute a problem whose solution is a net of interacting actions that when executed by the appropriate resources must attain the goals.

Planning addresses the last four stages of a decision making process, namely: design, choice, implementation and control. We have designed and implemented an architecture, called CHARADE, which integrates CBR with constraint satisfaction, and is aimed at supporting the user in the planning stage of a disaster management process. That technical solution will be described fully in the next section. Let's now go back to the application problem. Emergency management is a complex activity and professional firemen must acquire a vast amount of both theoretical and practical knowledge. That's normally provided in training sessions based on simulation. Firemen are put in a simulated situation and they must act, that is, take decisions and implement them. IT is becoming more and more important in supporting that training. Simulations performed in a real natural environment are costly and impossible to repeat. For that reason computer based simulations are becoming popular. In this context, the trainee sits in front of a computer system that presents him an emergency situation and asks him what actions should be taken. This starts a loop of action/feed-back, which is assisted by the computer simulation, and controlled by the trainee that aims at a solution stage.

We have developed a CBR system, named CARICA, that is used by the trainer to select and replay simulation exercises. It enables the user to browse with different modalities a case base of past forest fires that was acquired in a real fire campaign. The system further makes possible knowledge extraction from a case base in the form of statistics on the data, visual description of the data and selection of relevant prototypes. CARICA will be fully described in a next Section.

4. Planning the first attack

Planning the first attack to a forest fires means to match an emergency scenario with an integrated course of actions that when implemented in the real environment will not stop the emergency. Moreover, since many plans may achieve the posed goals, one further try to optimize, i.e. select the plan with minimal time span or minimal resources consumption.

In a classical AI perspective there is a neat distinction of roles between the user and the planner: the user poses the goal, the environment sets the initial conditions and the planner finds out the solution. This simplified view of the problem solving activity does not apply in our case. The user is an expert of the domain. He might be able to solve by himself the current goal, for example mostly regarding strategical decisions. In other situations he might want to set constraint to the search process. Decisions are always taken by the user, using the computer and other tools, usually following a complex operational flow. Therefore it is needed a system that could be used in a more flexible way. Assumptions may be done in any moment and change of focus of reasoning are continuous, for example new fire-alarms may arrive at any moment of the planning process.

The above mentioned issues stress various limitations of traditional AI planning techniques, in particular those developed in the context of pure formal approaches to planning. Driven by domain requirements and constraints we have defined a planning approach that rests on the integration of case-based reasoning and constraint reasoning.[3]

A simplified version of the CHARADE planner architecture is shown in Figure 1. This picture shows the main components: the case-based reasoner, which retrieves (stores) plans from (to) the plan memory; the plan space, which is the data space where plans are installed after retrieval; the action manager, that manages the user requests to modify the plans installed in the plan space; the constraint reasoner, in which variables and constraints are created and constraint reasoning algorithms called. In particular, the picture shows the constraint network representing the temporal structure of the plan: temporal variables of the plan and constraints defined on them, as presented in detail below.

The memory of plans is the case base of the CHARADE system. Each plan in fact merges two types of information: scenario description and network of actions, i.e. the conditions under which the plan was applied and the plan itself. The scenario description comprises more than forty features that characterize completely the emergency situation as it comes from the assessment phase. For example, the meteo condition of a case is described by features like: air temperature, air pressure, air humidity, wind direction, wind speed, etc. Similarly are described: the fire extension and intensity; the availability of the principal types of resources (airplanes, helicopters, water tanks, human squads,...); the prevailing vegetation of the fire area; the accessibility level for each type of available resource; and so on.

When a new emergency occurs, the situation is assessed and a new scenario description is built. Then the user calls the CBR to retrieve from the memory a small subset of cases that partially matches the current scenario description. The retrieval is performed by measuring the "distance" from the current case scenario description and that of a stored case. We are therefore assuming that similar scenarios must have similar plans (solutions). The cases with smallest distance are finally presented to the user. The key element of that nearest neighbor retrieval[19] is the distance function. CHARADE exploits a novel local distance function where the relevance of each feature is learned from the same case base using a feed-back learning procedure.[4] The word "local" means that the

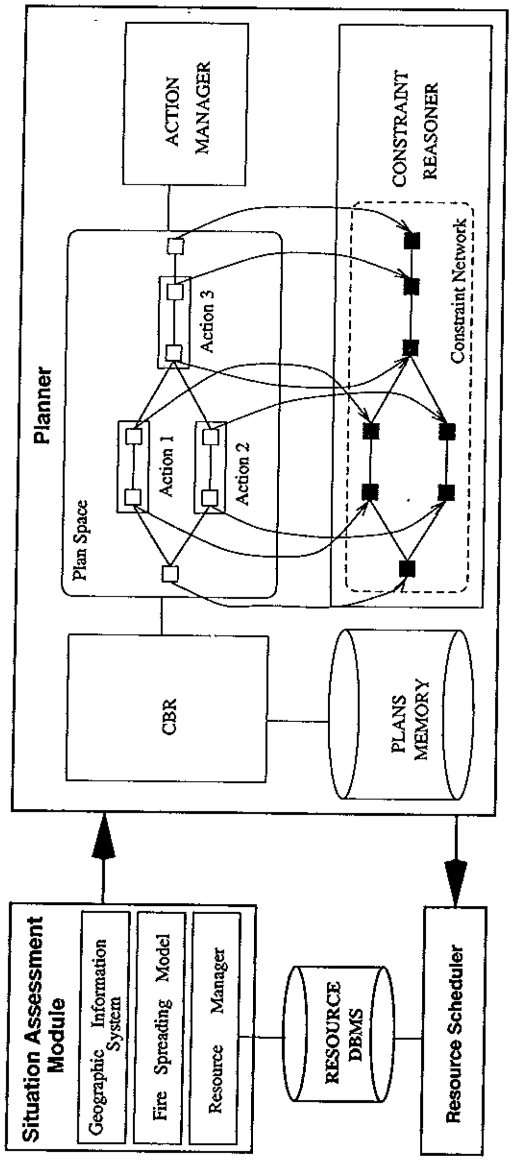

FIG. 1: *The architecture of the CHARADE planner*

definition of the metric changes with the input cases to be compared. Classical metrics do not have this behavior, the most widely used approach consists of computing feature by feature a "difference" and then summing up those differences in a total distance. For numerical features the difference in module is used and for nominal features the "difference" of two values is simply null if the values are equal and one otherwise. Feature relative relevance is considered by weighting the corresponding difference with a positive number that ranges between 0 and 1. In a local metric the feature weights are not constant values but depends on the cases to be compared. In that way the metric can model a condition like: the wind speed is maximally important when the humidity is low and the height of flames is high.

As stated above, the second component of a case is the network of actions. That network is the core component of the plan and represents the "solution" to the planning problem posed by the scenario description. When a plan is retrieved from memory by the case-based module the network of actions is copied on the dynamic data space. The temporal structure stored in memory (constraints between plan's components) is installed, that is a temporal constraint network is defined on the time variables occurring in the plan according to a script that is stored in the case. After that step, this data structure is fitted and modified to adapt it to the specific scenario data. Part of this process is done batch by the system (fitting), part is performed in collaboration with the user through the action manager (adaptation).

For example if a retardant bombing action is in the retrieved plan and the current fire area is one third larger than that of the retrieved case the estimated duration of this action must be prolonged proportionally. Or if a fire line action is performed on the right flank on an area that the mathematical spreading model estimates will be burned one hour after the fire ignition then that action must be completed before that deadline. The satisfiability of this set of constraints is checked by an incremental constraint propagation algorithm fully described elsewhere.[20,21] The temporal dimension of the plan is managed with a quantitative approach based on temporal variables over continuous domains.[22]

At this point, the user plays a more active role. A tentative plan for the current emergency situation has been retrieved and automatically adopted, but may still be unsatisfactory for a number of reasons. For instance, the case base may not contain a well-matching situation, or the user may be unaware of a good opportunity to exploit a set of resources that are moving back to their base but that can be redirected on the fire zone. It would be impossible to monitor all the more subtle elements that can motivate the final decision. The retrieved case, in our opinion, can only be a seed for the last crucial step of manual plan adaptation. In that phase the user can add new actions, discard some of those contained in the retrieved plan, pose new constraints or maybe start a totally new plan from scratch. That activity is supported by the CHARADE planner through the action manager module. Using a library of actions and constraints the user can edit the plan and the underlying constraint satisfaction system can quickly check the feasibility of the plan.

The last activity supported by CHARADE is automatic and manual allocation of resources (the relative modules are not depicted in Figure 1) finding correct schedule times for allocated actions and updating the feasible times for actions not yet allocated.

5. Mining cases

An extensive evaluation of the CHARADE system showed that the same case base can also be exploited for another user need, i.e. for *learning* how to plan initial attack to forest fires and more in general for acquiring knowledge from the data stored in the case base. For that reason, in a second project called CARICA we moved from a strict CBR paradigm to a Knowledge Discovery in Database (KDD) system.[5] KDD refers to the overall process of discovering useful knowledge from data, mapping low-level data into other forms, more compact, more abstract and more useful. A case base, a product of the CBR methodology, is a rich source of information that can be reused beyond the original problem-solving task, it can be "mined" with KDD techniques. CBR tools can be used for that goal, but some changes are needed. For instance, the retrieval algorithm, the primary component of a CBR system, must discover similar case, but similarity must be redefined query by query according to the target information searched by the user. The reusable component of the CARICA project is called CBET (Case-Base Exploration Tool) and is devoted to mining case bases.[23]

In CBET many case bases can be loaded simultaneously, enabling parallel exploration of different knowledge sources. CBET supports editing of the structure of the case base like deleting a feature or creating a new feature whose values are obtained by the application of an arbitrary function of other features.

A case base can be browsed in two modalities: table and features. The table modality shows a table where the columns display features (values) and the rows are cases. (In Figure 2, 'ID' and 'code' are shown.) The user can select the features to show and the order in which they must appear on the table. That enables a simple customization of the interface.

In the second modality all the feature descriptors are listed. This provides another perspective to the information contained, which would help, for example, in selecting the most appropriate features to show in the table modality. A closer look to a feature is provided by another panel

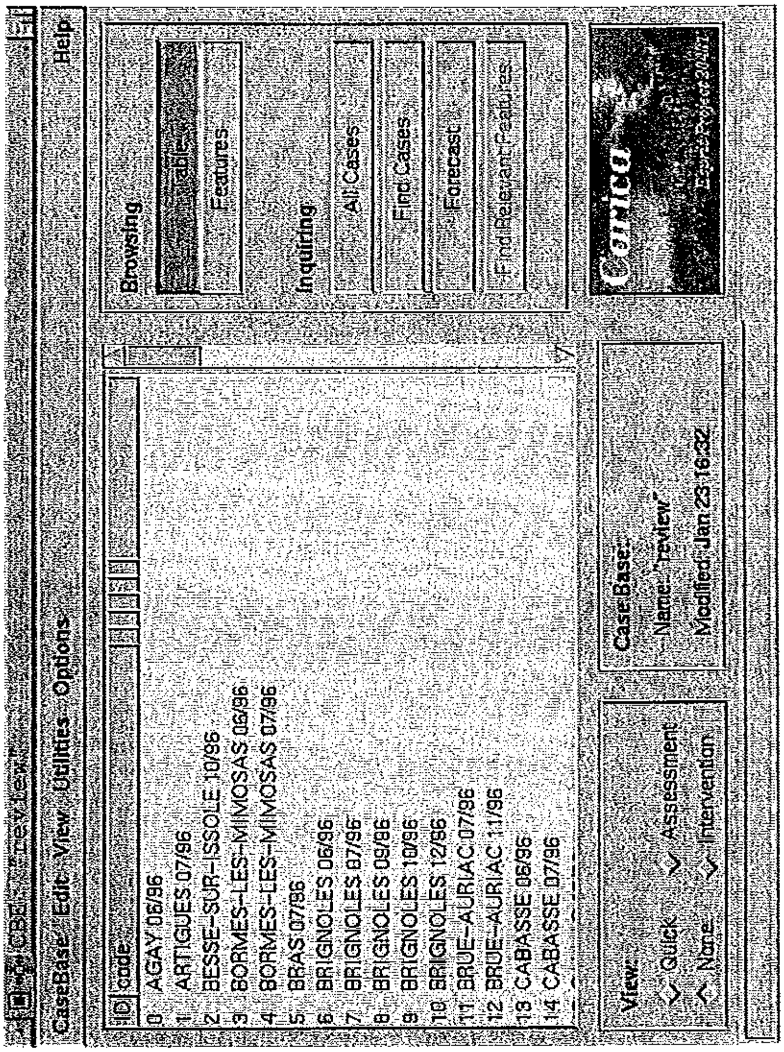

FIG. 2: *The table browsing modality*

that shows summary information on the feature: maximum value, minimum value, type, the presence of missing values, mean, median, etc.

The system can also show summary information on a selected case (see Figure 3). In the forest fire application this panel shows a simple map of the zone surrounding the fire, the location of the fire and the description of the wind direction and speed. It also shows the values of the 'surface', 'meteo risk' and 'vegetation' features and a pie chart of the types of resources used in the intervention.

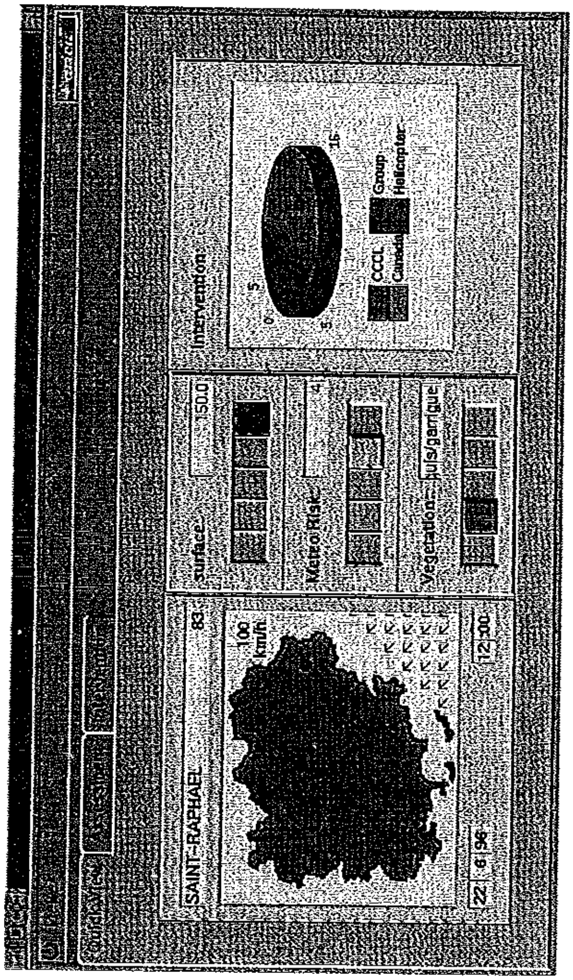

FIG. 3: *The quick view panel*

Many different types of plots can be shown with CBET. For example, simple feature histograms can show the distribution of both numerical and nominal features. Two-dimensional plots of numerical features can also be shown. When one of the two features to plot has nominal values, CBET shows the distributions (histograms) of the numeric feature values conditioned to the different values of the nominal feature. It produces a set of histograms (one for each different value of the nominal feature) that show the influence of the nominal feature on the distribution of the numeric feature. Finally, if two numeric features are plotted, different icons can show a third nominal feature.

CBET can show dependencies between data contained in a case base, i.e. how the variation of a feature can impact on another. The user must select the target feature (the feature he/she is interested in) and the algorithm to be used for evaluating the relevance of the other features in describing the behavior of the target. We hypothesize that the features that maximize the 'purity' objective functions used for top down

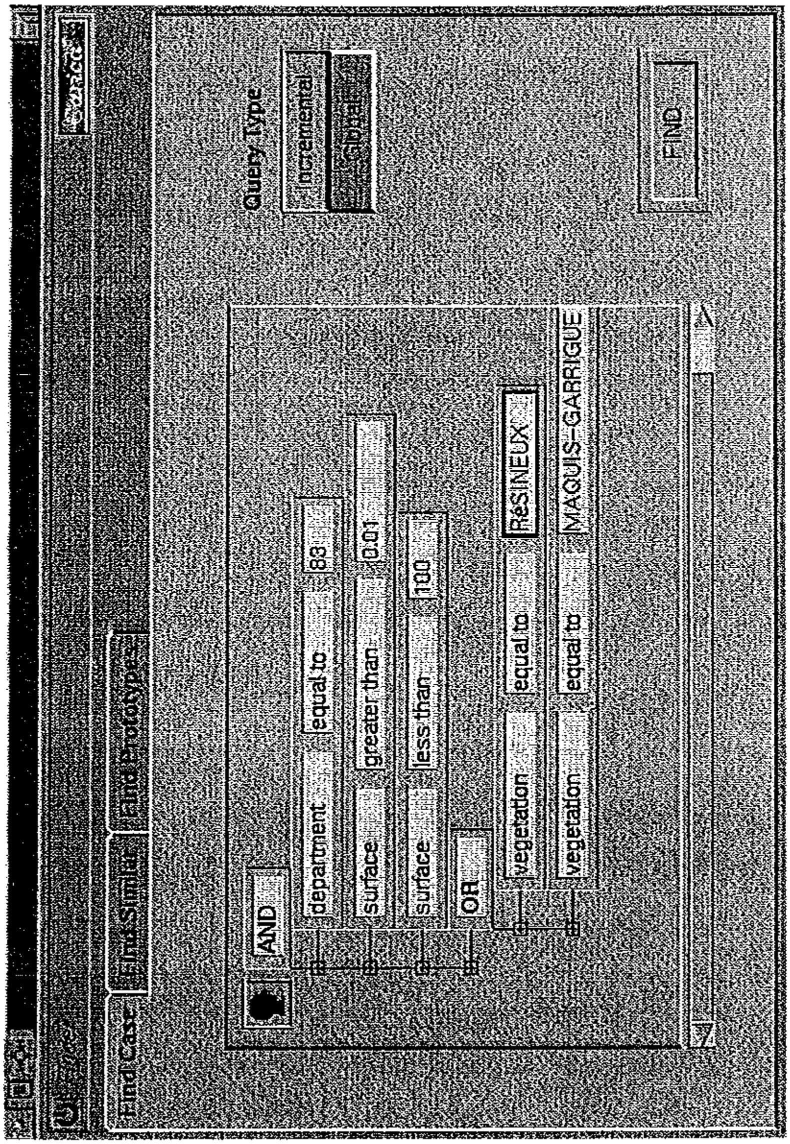

FIG. 4: *A query by example*

induction of decision trees[24] can also be valuable for evaluating the relevance in our application. The result can be viewed in two ways: as an ordered list of features, or with a bar plot showing for each feature the precise value of the relevance. The evaluation of the relevance of features can also be stored in the case base, enriching its information content.

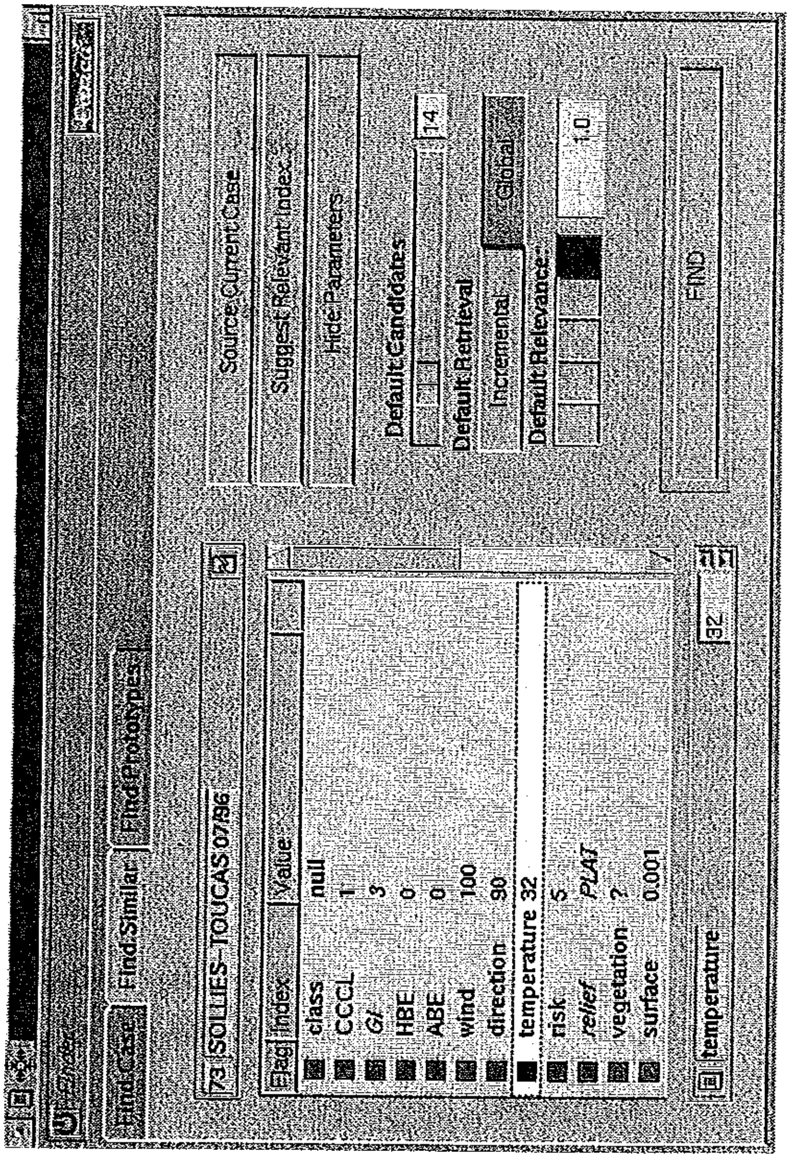

FIG. 5: *Changing the relevance of the 'temperature' feature in searching for a similar case*

In a similar way the relevance of pairs of features can be estimated. That provides a heuristic for selecting those pairs of features likely to produce the most clear and effective plots.

CBET searches a case base in two ways: the classical query by example and the case base search. Queries by example like that depicted in Figure 4 can be easily composed.

A second type of search is the "fuzzy" search performed by a nearest neighbor algorithm with feature relevance computed as described before. Every case can be selected as probe to search for similar cases. Some feature values of the selected case and the feature relevance computed by the learning algorithms quoted above can also be modified, therefore building a completely new case (probe) (see Figure 5).

Searching for prototypes is another important function supported by CBET. A prototype is a case that has a cluster of similar cases in the case base. These similar cases can be described by referring to this prototype. The prototypes created can be loaded and the partition of cases in the clusters represented by prototypes can also be shown to the user.
The last collection of functions in CBET is those related with forecasting of unknown feature values. In order to solve this problem a classifier is built and then the classifier is used for evaluating the most likely value of the unknown feature. CBET supports many types of classifiers, including: k-NN and Instance Based classifiers,[6] C4.5,[24] NGE.[25]

6. Related work

Case-Based planning is a vast area of research originated from a pioneering work of Hammond[26] and ranging above many different approaches.[27] A remarkable work in combining different specialized AI techniques (such as CBR) in dealing with real planning problems is ongoing within the ARPA & ROME Laboratory Knowledge Based Planning Initiative since many years (partially described in).[28]

Two very recent approaches that are strictly related to our both in the techniques and in the application domain have been presented recently at AAAI-98 workshop on Case-Based Reasoning Integrations. First, Gervasio et al.[29] propose to use the case library to provide an initial candidate solution that the human user is then free to adapt to suit the current situation. The type of emergencies that they consider is spill of some material with hazardous properties. They stress the hierarchical structure of a plan but plan adaptation is only performed manually without any use of a constraint propagation algorithm.

The second has been proposed by Leake et al.[30] Their domain is disaster response planning and they combine case-based reasoning with rule-based reasoning. Rather than simply switching among the two techniques when one is more appropriate than the other the authors shows how each single technique can support the other. For instance the case-based reasoner can call on the rule-based reasoner as it refines a retrieved case.

Another approach very similar to our is described by Lee et al.[31]. The authors are aimed at generating apartment construction project networks. The system (a fully implemented application) reuse previous cases for planning a new project. They combine in a plan the physical layout with a project scheduling. They augment network-based project-planning techniques with knowledge of physical constraints and past experience, represented as cases.

7. Conclusions

We believe that case-based reasoning techniques can be the key factor in dealing with complex environmental problems like natural disaster. CBR can support both the "reaction" to the emergency, when is integrated in the planning component, and the knowledge extraction task when is the core engine of a data mining tool. CHARADE and CARICA are two successful (Esprit) projects that have practically demonstrated the usefulness of CBR in that domain. CHARADE and CARICA were developed in collaboration with a department of firemen of the southern France, in a consortium of both industries and research centres, and have been integrated into the range of user tools for daily use. More information on these projects can be found at:

http://mnemosyne.itc.it:1024/avesani/html/charade.html
and
http://mnemosyne.itc.it:1024/avesani/html/carica.html.

8. References

1. RICCI, F., MAM, S., MARTI, P., NORMAND, V. *and* P. OLMO. *CHARADE: a platform for emergencies management systems. Technical Report 9404-07*, IRST, 1994.
2. AAMODT, A. *and* E. PLAZA. Case-based reasoning: foundational issues, methodological variations, and system approaches. *AI Communications*, 7(1), 1994, 39-59.
3. KUMAR, V. Algorithms for constraint-satisfaction problems: A survey. *A.I. Magazine*, 13(1), 1992, Spring, 32-44.
4. RICCI, F. *and* P. AVESANI. Learning a local similarity metric for case-based reasoning. *International Conference on Case-Based Reasoning* (ICCBR-95), Sesimbra, Portugal, Oct. 23-26, 1995.
5. FAYYAD, U., PIATETSKY-SHAPIRO, G. *and* P. SMYTH. From data mining to knowledge discovery in databases. *AI Magazine*, 1996, Fall, 37-54.
6. AHA, D. W., KIBLER, D. *and* M.K. ALBERT. Instance-based learning algorithms. *Machine Learning*, 6, 1991, 37-66.
7. WETTSCHERECK, D., MOHRI, T. *and* D.W. AHA. A review and comparative evaluation of feature weighting methods for lazy learning algorithms. *AI Review Journal*, 11, 1997, 273-314.

8. KRIEGSMAN, M. *and* R. BARLETTA. Building a case-based help desk application. *IEEE Expert Intelligent Systems and their Applications*, 1993.
9. ALLEN, B. Case-based reasoning: business applications. *Communication of ACM, 37*(3), 1994, 40-2.
10. KETTLER, B.P., HENDLER, J. A., ANDERSON, W. A. *and* M. P. EVETT. Massively parallel support for case-bases planning. *IEEE Expert*, 1994, 8-14.
11. GEBHARDT, F., VÖSS, A. GRÄTHER, W. *and* B. SCHMIDT-BELZ. *Reasoning with complex cases*. Kluwer, 1997.
12. ASHLEY, K. D. Reasoning with cases and hypotheticals in hypo. *International Journal of Man-Machine Studies*, 34, 1991, 753-96.
13. ALTHOFF, K.-D., AURIOL, E., BARLETTA, R. *and* M. MANAGO. A review of Industrial Case-Based Reasoning Tools. *AI Intelligence*, 1995.
14. WATSON, I. *Applying Case-Based Reasoning: Techniques for Enterprise Systems*. Morgan Kaufmann, 1997.
15. WILSON, D. R. *and* T.R. MARTINEZ. Improved heterogeneous distance functions. *Journal of Artificial Intelligence Research*, 11, 1997, 1-34.
16. PORTINALE, L. *and* P. TORASSO. On the usefulness of re-using diagnostic solutions. *In Proceedings of the 12th European Conference on AI - ECAI 96*, Budapest, 1996, 137-41.
17. SIMON, H. A. A behavioural model of rational choice. *Quaterly Journal of Economics*, 69, 1955, 99-118.
18. ALLEN, J. F. HENDLER, J. *and* A. TATE. *Readings in planning*. Morgan-Kaufmann, San Mateo, CA, 1990.
19. DASARATHY, B. V. *(ed), Nearest neighbor (NN) norms: NN pattern classification techniques*. IEEE Computer Society Press, Los Alamitos, CA, 1991.
20. PERINI, A. *and* F. RICCI. An interactive planning architecture. In M. Ghallab and A. Milani, editors, *New directions in AI Planning*, 1996, 273-83, IOS Press.
21. GEREVINI, A., PERINI, A. *and* F. RICCI. Incremental algorithms for managing temporal constraints. *In IEEE International Conference on Tools with Artificial Intelligence* (ICTAI'96), 1996, 360-3, IEEE Computer Society Press.
22. DECHTER, R. MEIRI, I. *and* J. PEARL. Tèmporal constraint networks. *Artificial Intelligence*, 49, 1991.
23. AVESANI, P., PERINI, A. *and* F. RICCI. Cbet: a case base exploration tool. In Springer-Verlag, editor, *Fifth Congress of the Italian Association for Artificial Intelligence* (AI*IA 97)}, Roma (Italy), 1997, September 16-19.
24. QUINLAN, J. R. C4.5: *Programs for machine learning*. Morgan Kaufmann, San Mateo, CA, 1993.
25. SALZBERG, S. L. A nearest hyperrectangle learning method. *Machine Learning, 6*, 1991, 251-76..
26. HAMMOND, K. J. Case-Based Planning: *Viewing Planning as a Memory Task*. Academic Press, Boston, 1989.
27. VELOSO, M., MUNOZ AVILA, H. *and* R. BERGMANN. Case-based planning:selected methods and systems. *AI Communications, 9*(3), 1996, 128-37.
28. WILKINS, D. E. *and* R.V. DESIMONE. Applying an ai planner to military operation planning. *In* M. Zweben *and* M. Fox, *editors, Intelligent Scheduling*. Morgan-Kaufmann, San Mateo, 1992.
29. GERVASIO, M., IBA, W. *and* P. LANGLEY. Case-based seeding for an interactive crisis response assistant. *In Proceeding AAAI-98 Workshop on Case-Based Reasoning Integrations*, 1998.
30. LEAKE, D. *and* A. KINNEY. Integrating cbr components within a case-based planner. *In Proceeding AAAI-98 Workshop on Case-Based Reasoning Integrations*, 1998.

31. LEE, K. J., KIM, H. W., LEE, J. K. *and* T. H. KIM. Case- and constraint-based project planning for apartment construction. *AI Magazine, 19*(1), 1998, 13-24.

This paper was published in
THE NEW REVIEW OF APPLIED EXPERT SYSTEMS
Volume 5, 1999
Subscription price for Volume 5, 1999: £70.00/US$130.00
Published annually. Back issues available.
Taylor Graham Publishing, 48 Regent St, Cambridge CB21FD,UK
www.taylorgraham.com