PROBLEM SOLVING AS INTELLIGENT RETRIEVAL FROM DISTRIBUTED KNOWLEDGE SOURCES

Zhengxin Chen

Department of Computer Science,
Louisiana State University,
Baton Rouge, LA 70803

Abstract. *We propose a model of problem solving by dynamically distribute the knowledge sources to several processors in a controlled manner. Example is given, the features of this approach are also summarized.*

Introduction

Recently, intelligent distributed systems have drawn much attention. Researches in distributed artificial intelligence (DAI) have focused on cooperative solution of problems by a decentralized and loosely coupled collection of knowledge sources (KSs), each embodied in a distinct processor node [18]. Most previous works in DAI deal with distributed problem solving techniques, for instance, the investigation of phases of problem decomposition, sub-problem distribution, sub-problem solution, and answer synthesis [16]. In this paper we investigate distributed computing in intelligent systems from a different perspective. From the viewpoint that problem solving can be viewed as intelligent knowledge retrieval, we propose the use of distributed knowledge sources in intelligent systems. Owing to space limitation, no technical detail is given in this paper.

A model that integrates knowledge

We start from a cognitive model for knowledge retrieval reported earlier [2,3]. Information chunks or pieces (will be referred to as documents) are acquired, mapped into internal structure and integrated into an overall knowledge base, while the documents (which form the sources of the knowledge) are still identifiable. Notice that this model is very general. The documents may be either English-like texts or numerical data sets, and may have quite heterogeneous structures. The mapping mechanism may also vary a lot. For instance, it may be a natural language understander to "understand" the natural language-like input or a kind of data analyzer to analyze the input numerical data. These internal structures (i.e., the result of the mapping) are referred to as knowledge, integrated to an overall knowledge base, and can be retrieved. This kind point of view is consistent with the view that knowledge is condensed information [15]. After knowledge is retrieved, they will be presented in a easily readable form (e.g., by reconstructing the documents) to the user.

Problem solving and knowledge retrieval

The central idea of this report is to relate problem solving to knowledge retrieval. This is a topic which needs further investigation, although it is not new. In fact, the relationship between information retrieval and question-answering system which has been discussed by many authors is basically also true for the relationship between knowledge retrieval and problem solving. According to [8], systems having broad, possibly interrelated data bases whose answer-computation mechanisms is not capable of great depth tend to be called question-answering systems while systems having less-interrelated data bases whose answer-computation mechanism is capable of more depth tend to be called problem-solving systems. Based on this understanding, if a question-answering system is a kind of information retrieval system that understands the texts, it is reasonable to say that a problem-solving system may be realized as a kind of knowledge retrieval system which needs in-depth understanding and handling of the knowledge. Procedurally, a problem solving system utilizes the knowledge in a manner which results in a sequence of retrieval steps. The objective of the problem solving system is to make decisions to identify and integrate certain parts of knowledge for certain goal(s) and actually use the related knowledge in an intelligent way. The tasks of the decisions are to make a coherent final plan or to integrate various partial solutions, to name a few.

The use of distributed knowledge sources

What is more, frequently it is desirable to retrieve knowledge from more than one knowledge source (KS). For instance, operational system exists in space science which is able to combine evidence from multiple sources [1]. But along with this direction, much work is still ahead. This particularly includes to develop a useful control mechanism to make this scheme work systematically. The rationale of using our model for this purpose can be justified as below. It has been recognized that sufficiency of knowledge is one of the most important requirements in generating some sequence of partial interpretations that culminates in correct complete interpretation [11,12]. In case of lacking the proper tool of handling the entire knowledge at once, we may try to distribute the entire knowledge into several smaller knowledge sources, each of them can be handled by an independent processor. The various knowledge sources serve various documents in our model; the task of intelligent retrieval is to capture the underlying meaning of these knowledge sources handled by the processors. Each knowledge source provides part of the knowledge needed to solve the problem; therefore, an additional task involved in the problem solving process is to control these processors and to force convergence of the solution, and an overall solution based on all the partial solutions can thus be finally obtained.

By "intelligent retrieval" we mean that (1) the problem solver deals with the "internal form" (or meaning) of the knowledge sources, not necessarily its original form; (2) the problem solver is able to use its rule base to handle the partial or conflicting information obtained from different knowledge sources. Therefore, even though each node has only a limited view of the input data, the problem solver is able to integrate the partial solutions and to convergent to a final solution.

The architecture of our problem solver is explained in Fig. 1. The conceptual memory serves a role of index of the knowledge sources; it is used in integration knowledge from different knowledge sources as well as retrieval of these knowledge sources. The rule base provides rules for integration of knowledge sources.

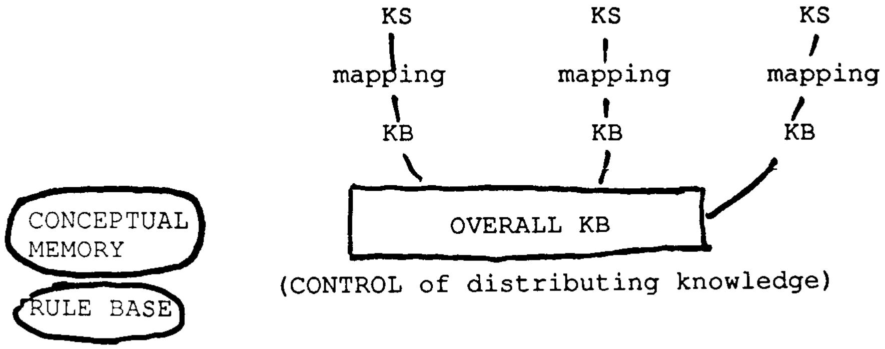

Fig. 1

The fundamental idea is going to be illustrated as below. Previously we described in model which concerns the problem of generating a plan to access heterogeneous numerical database dealing with observational data. In the following we consider another application, which concerns qualitative reasoning on partial results obtained from distributed quantitative processors (part of this work was reported in [5]). This second application, which handles the conflicting information obtained from partial solutions, is more intersting.

This approach utilizes the original model in a "reversed" manner. Traditionally, in a knowledge-based system, knowledge is acquired first, and serves as the environment of solving the problems. In our approach, *input data are treated as knowledge source,* and, consequently, solving this specific problem means to understand the data, i.e., to *intelligently retrieve the knowledge* implied by these data. This also means to distribute data (instead of problem) and assign them as knowledge sources to processors in a *controlled* manner. The function of the processors is to process (or map) them into internal form (or "knowledge").

Our approach is somewhat related to the work of distributed Hearsay-II [11,12], in which a distributed approach of problem solving has been investigated. An interpretation system accepts a set of signals from some environment. Two major questions are how to interpret the data and how to decompose a given interpretation technique for distribution. It is necessary to operate on local databases that are incomplete and possibly inconsistent and to integrate incomplete partial solutions to construct an overall solution. The elimination of explicit synchronization has increased parallelism. Our approach shares some common features with these previous approaches, the difference is only at what is to be decomposed or distributed.

To illustrate, let us consider the solving of the following problem. This problem solver deals with periodically collected observational numerical data which involve a lot of variables. Only one of the variables is considered as system function (dependent variable), the others are treated as independent variables (although they may be somewhat interrelated). The problem is to find, among a large set of independent variables, the most important variables which effect the system function. Algorithms exist to deal with a limited amount of variables, and they can be actually carried out by existing software (for instance, the technique of utilizing entropy data analysis introduced by [9]). Since each time only a limited set of variables can be considered, each time we can only obtain a partial solution. The type of problem discussed in this paper is similar to the data compression schemes for inertial navigation systems discussed in [13], in which frequent data are collected while the computation capability is limited, but the techniqued used here is entirely different.

For this particular problem, our scheme of solving problem through retrieval of distributed knowledge sources can be explained as follows. Data are decomposed into several subsets, each is able to be handled by a single processor. The part of data distributed to a processor (in our current example, in addition to the dependent variable, each decomposed data set includes several independent variables), is viewed as knowledge source associated with it. (The knowledge sources are not necessarily disjoint). Each process can treat its own knowledge source either as a single unit or a set of knowledge sources at lower levels. All these processors can work on its own knowledge source simultaneously and find the most important variables based on this knowledge source. As the result of this processing (or "mapping") is a set of rules which reflect the knowledge implied by this particular set of data. Each assumes its knowledge source is the only existing knowledge to the system, and claims the variables it found are the dominant ones to the whole system. Under this architecture, the type of problem to be solved can be restated as follows: given a set of data which are distributed to the KSs (with arbitrary number), how to determine the limited number (say, 4) of dominant variables from the results of the competing processors?

Basically, our problem solver solves this problem in the following manner. A set of rules maintained in the central node is used to integrate the intermediate results obtained from the processors. Integration includes to handle the conflicting information and draw similarities among the partial solutions provided by the independent processors. A few new sets of data which includes reduced number of variables are thus created; they are treated as knowledge sources and are then assigned to several processors. The number of variables remained in the knowledge sources are thus reduced and finally they are convergent to the solution. There is a centralized control over the knowledge source the processor possesses.

The problem can be solved by following those steps:

1. Decompose the input data into several subsets, each consists several independent variables and the dependent variable. These subsets form the distributed knowledge sources, each of them is associated with processor(s) which is(are) able to process the associated knowledge source in some way. In addition to this kind of decomposition, a set of rules also exists so that these partial solutions may be integrated later by these rules.

2. Retrieve knowledge processed by processors, use rules to corporate information and get rid of conflicting information.

3. Form reduced knowledge sources, and assign back to some processing nodes.

4. Repeat steps 1-3 until a convergent solution set is obtained.

Notice that in step 1 all the processors related to knowledge sources are not necessarily homogeneous. But to simplify the discussion, in the following example, we will assume all the processors take the same form. Notice also that since the number of variables to be considered at each iteration is at least decreased by one, our scheme can guarantee the convergence of the solution, although this does not necessarily means optimal at all (see the conclusion part of this paper).

To illustrate, suppose we have the original data including variables A, B, C, D, E, F (Fig. 2a), but processor is able to handle up to four variables at each time. Suppose based on domain related knowledge, it is able to organize knowledge sources in a way shown in Fig. 2b. After processing these knowledge sources parallelly, it is possible to identify the partial solutions obtained from these knowledge sources, and rules may be used to form "better" knowledge sources which only includes variables A, B, C, or A, B, E. Therefore only variable combination A, B, C, E needs to be considered. The size of the solution set is thus reduced. The final solution of the original problem can be found by processing this data set.

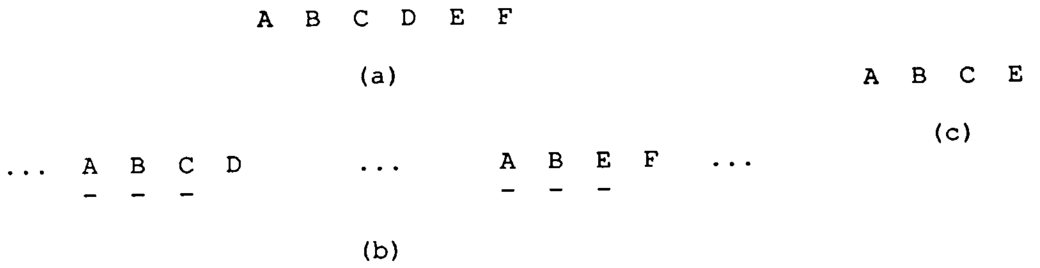

Fig. 2

Features and comparisons with other works

Usually, in distributed problem solving, a single task is envisioned for the system, while distributed processing systems synthesize a network which is able to carry out a number of widely disparate tasks. Since our system is aimed to solve one single task at one time, it is close to distributed problem solver, but the control in our system is not decentralized. Briefly, our scheme has the following features:

(1) Deliberately distribute input data as knowledge sources rather than decompose the task.

(2) Centralized control is only restricted at each knowledge source level.

(3) The problem is solved gradually by reducing knowledge sources, there does not exist a separate phase of answer synthesis.

The system described in this paper may be referred to as distributed knowledge processing system. Although the majority works related in distributed problem solving deal with knowledge sources which cooperate in the sense that no one of them has sufficient information to solve the entire problem [17], our scheme is the only one which controls the distribution of the knowledge sources (instead of the problems) in a dynamic manner. This is the fundamental difference between our approach and the others.

A systems level approach to distributed processing was suggested in [14], in which a scalable, dynamically reconfigurable architecture was claimed to be necessary. This means a computer with no architecturally imposed performance limits. If this approach is to find a hardware solution, then our purpose is to find a software solution for a similar problem.

Integrating knowledge sources for computer "understanding" tasks was discussed by [6]. Our scheme is similar to that scheme which is a system of cooperating experts running in separate images. But ours is aimed to be a general knowledge integration scheme extended from the existing IR model, and is not restricted to text (written in English) understanding. Therefore, in this sense, ours is more general.

Concluding remarks

The method introduced in this paper does not necessarily generate the "optimal" solution; but, it does provide an acceptable one. We have successfully used the method described in this paper to find the effect of some most important variables to the system function [5]. Moreover, since the method introduced in this paper involves symbolic (qualitative) reasoning attached to numerical processors, it may be viewed as an example of coupling symbolic and numerical computing [10], which has been recently more and more discussed in space science as well as many other research fields.

References

[1] Campbell, S. D. and S. H. Olson, "Recognizing low-altitude wind shear hazards from doppler weather radar: an artificial intelligence approach," *J. Atmospheric and Oceanic Technology,* vol. 4, No. 1, March 1987, pp. 5-18.

[2] Chen, Z., "Some aspects of a cognitive model for information retrieval," *Proc. 18th Pittsburgh Conf. on Modeling and Simulation,* 1987.

[3] Chen, Z., "A language for modeling users virtual machine of information retrieval," *Proc. 1987 IEEE Workshop on Languages for Automation,* 1987.

[4] Chen, Z., "PENDS: a prototype expert numeric database system," paper presented at 2nd AIRES Workshop (AI Research in Environmental Science), 1987.

[5] Chen, Z. et al., "Qualitative reasoning for numerical systems," paper presented at 2nd AIRES, 1987.

[6] Cullingford, R., "Integrating knowledge sources for computer 'understanding' tasks," *IEEE Transactions on Systems, Man and Cybernetics,* vol. SMC-11, No.1, Jan. 1981, pp. 52-60.

[7] Davis, R. and R. G. Smith, "Negotiation as a metaphor for distributed problem solving," *Artificial Intelligence,* 1983, pp. 63-109.

[8] Green, C., *The Application of Theorem Proving to Question-Answering Systems,* Garland Publishing, New York, 1980.

[9] Jones, B. "Reconstructability considerations with arbitrary data", *Int. J. Gen. Sys.,* vol. 12, 1986, pp. 1-6.

[10] Kowalik, J. S. (ed.), *Coupling Symbolic and Numerical Computing in Expert Systems,* North Holland, 1986.

[11] Lesser, V. R. and D. D. Corkill, "Functionally accurate, cooperative distributed systems," *IEEE Transactions on Systems, Man, and Cybernetics,"* vol. SMC-11, No.1, Jan. 1981, pp. 81-96.

[12] Lesser V. R. and L. D. Erman, "Distributed interpretation: a model and experiment," *IEEE Transactions on Computers,* vol. C-29, No. 12, Dec. 1980, pp. 1144-1163.

[13] Medan, Y. and I. Y. Bar-Itzhack, "Batch recursive data compression schemes for INS error estimation," *IEEE Transactions on Aerospace and Electronic Systems,* vol. AES-21, No. 5, Sep. 1985, pp. 688-697.

[14] Meier, R. J. Jr., "A systems level approach to distributed processing," *Computer Sciences and Data Systems* (Proceedings of a symposium), NASA Conf. Pub. 2459, vol. 2, pp. 143-172.

[15] Michalski et al.(ed.), *Machine Learning: An Artificial Intelligence Approach,* Vol. 2, Kauffman, 1985.

[16] Smith, R. G. and R. Davis, "Framework for cooperation in distributed problem solving," *IEEE Transactions on Systems, Man, and Cybernetics,* SMC-11, No. 1, Jan. 1981, pp. 61-70.

[17] Smith, R. G., "Report on the 1984 distributed artificial intelligence workshop," *AI Magazine,* Fall 1985, pp. 234-243.

[18] Smith, R. G., "The 1985 workshop on distributed artificial intelligence," *AI Magazine,* Summer 1987, pp. 91-97.