Hans-Albert Schneider  
Universität Kaiserslautern  
Fachbereich Informatik  
D-6750 Kaiserslautern

Werner Dilger  
FhG-IITB  
Sebastian-Kneipp-Str. 12  
D-7500 Karlsruhe

ABSTRACT

Associative processors keep their data in content addressable memories (CAMs) accessing them by content, not address. Our interest concentrates on how associative processors can be used for inference processes. We have developed the model of an associative processor based on the Deduction Plan theorem proving method. Our approach also includes a unification algorithm which provides information about all causes of unification conflicts (if any) and allows simple backtracking of the unification graph.

1. Introduction

The idea to store data in memory cells which are accessed by their contents instead of their addresses is rather convenient because this is more similar to the way humans access data in their mind. Memories with this property are called content addressable memories (CAMs for short) or associative memories. There are different models of CAMs which differ in their degree of parallelity and the operational abilities of their cells.

In a restricted sense, associativity is achieved in high level programming languages by the ability to choose expressive identifiers. However, it often is necessary to access e.g. the elements of a list not as "the fifth entry in list PERSONS" but as "the entry in PERSONS where NAME='Smith'". This normally results in a lot of search time and/or management overhead, and the attempt to find a good compromise between both is classically known as time/space tradeoff. CAMs, however, can deal with this problem by operating on all cells (or a selected subset of cells) in parallel.

Although the idea of CAMs came up in the middle fifties, there are only few applications where they are used.

* Project sponsored by the Deutsche Forschungsgemeinschaft (DFG) under grant number Ma 581/4-1
The property to access cells by content makes CAMs well suited for applications where pattern directed retrieval plays an important role. There are a lot of pattern directed deductions in artificial intelligence (AI), thus associative processors (processors that store their data in a CAM) seem to have a good application domain there. We have picked out a special area of AI, namely automated theorem proving, to develop the model of an associative processor for it.

This paper is organized as follows: The second section gives a brief overview on the structure of CAMs and associative processors. The third section then sketches the theorem proving method we use, and the fourth section gives some idea of the model and how it would operate.

2. Associative Processors

For an overview on CAMs see also [4].

A content addressable memory is a storage unit that stores data in a number of cells such that those cells can be accessed by their content. The smallest unit of a CAM is the bit cell. It is able to

1) store one bit of information
2) read out one bit of information
3) compare its content with given information.

Searching is done using masks and the comparison operations of the cells.

An associative processor is a processor that uses a CAM as its data store. It therefore essentially consists of

1) a CAM
2) an arithmetic and logic unit (ALU)
3) a control system with an instruction store.

3. Deduction Plans and UwC

For the standard definitions of the terms used in automated theorem proving, such as clause, literal, or term, cf. [1].

The deduction plan theorem proving method is a graph based approach, that is, the proof is done by manipulation of a graph. This inference graph will be called a deduction plan throughout this paper (however, the terminology is slightly different in [3]). Its nodes are variants of input clauses (i.e. sets of literals). Its edges are directed and labelled by triplets \((t,u,v)\) where \(u\) and \(v\) are the literals of interest of the incident clauses \(u\) of the starting