1.

Title: BELIEF MAINTENANCE IN DYNAMIC CONSTRAINT NETWORKS

Author:

Rina Dechter (Cognitive Systems Laboratory Computer Science Department University of California, Los Angeles, CA 90024)

Avi Dechter (Department of Management Science California State University, Northridge, CA 91330)

2.

Source: DBLP

Date of publication: January 1988

Link: http://www.ics.uci.edu/~csp/r5.pdf

3.

A dynamic Constraint-Network (DCN) is a sequence of static CNs each resulting from a change in the preceding one, representing new facts about the environment being modeled. As a result of such an incremental change, the set of solutions of the CN may potentially decrease(restriction) or increase(relaxation).

Restrictions occur when a new constraint is Imposed on a subset of existing variables, or when a new variable is added to the systems via some links. Restrictions always expand the model, ex… they add variable and constraints so that the associated constraint graph grows monotonically.

Relaxation occurs when constraints that were assumed to hold are found to be invalid and, therefore, may be removed from the network. However, it is not necessary to remove such constraints in order to cause the effect of relaxation. This can be achieved by modeling each potentially able to relax constraint in a special way which involves the inclusion of a bi-valued variable whose values indicate whether the constraint is active or not. Thus, we may assume, as is common in truth maintenance systems, that constraints that are added to the system are never removed.

4.

The propagation scheme contains two components: (1) support updating and (2) contradiction resolution. The fist handles noncontradictory inputs and requires one pass through the network. The second finds a minimum set of assumption changes which resolve the contradiction. Contradiction resolution may take five passes in the worst case: activating a diagnosis subtree (one pass), determining a minimum assumption set (two passes) and updating the supports with new assumptions (two passes).

Support propagation, assume the network is initially in a stable state, namely, all support vectors reflect correctly the constraints, and that task is to restore stability when a new input causes a momentary instability. The updating scheme is initialized by the variable directly exposed to the new input. Any such variable will recalculate and deliver the support vector for each of its neighbors. When a variable in the network receives an update message, it recalculates its outgoing messages, sends them to the rest of its neighbors, and at the same time updates its own support vector. The propagate through the network only once (no feed-back), since the network has no loops. If the new input is a restriction, then it may cause a contradictory state, in which case all the nodes in network will converge into all zero support vectors.

Contradiction resolution, when, as a result of new input, the network enters a contradictory state, it often means that the new input is inconsistent with the current set of assumptions, and that some of these assumptions must be modified in order to restore consistency. We assume that certain variables which initially are assigned their default values but may at any time assigned other values as needed. The task of restoring consistency by changing the values assigned to a subset of assumption variables is called contradiction resolution.

Once contradiction resolution has terminated, all assumptions can be changed accordingly, and the systems can get into a new stable state by handling those changes using support propagation. If this last propagation on the network may be proportional to the number of assumptions changed. If, however, these message updating is synchronized, the network can reach a stable state with at most two message passing on each arc.