8-15-1997

Visualizing the Knowledge Base of a Strategic Planning Expert System

Maurice Danaher
Edith Cowan University, m.danaher@cowan.edu.au

Visualizing the Knowledge Base of a Strategic Planning Expert System

Maurice Danaher
Department of Computer Science, Edith Cowan University,
Mt Lawley, Western Australia 6050
Email: m.danaher@cowan.edu.au, Fax: +61 9 370 6100

Abstract

A knowledge base can be very large containing many hundreds of objects and relationships. A graphical representation can assist in exploring and understanding the structure. The work discussed here is concerned with visualizing the knowledge structure in an expert system that has been developed for strategic planning. The method involves determining an initial layout for the structure followed by application of a fisheye viewing technique. Here we outline the approach taken to produce an initial layout of this complex structure.

Introduction

A graphical representation is an effective means of communicating relationships between objects and it facilitates exploration of the structure of the knowledge base. Techniques have been proposed for visualizing hierarchical and linear information structures (Mackinlay et al, 1991; Robertson et al, 1991) but these approaches are not suitable for knowledge bases which are arbitrary in nature.

In an expert system, SMP, that has been developed for strategic planning in manufacturing, knowledge about an enterprise is represented by using two fundamental concepts - objects and relations. To display the SMP knowledge base in the form of a graphical network, an appropriate layout must be determined. In this work the objects are represented as labelled rectangles (referred to hereafter as *nodes*) and the relations simply as lines (referred to hereafter as *arcs*). Due to the arbitrary nature of the SMP knowledge base, there can be a large number of arc crossings in the graphical network. These intersections present an obstacle to comprehending the structure of the knowledge base.

Unfortunately there does not appear to be any general solution to the question of optimally positioning nodes such that the knowledge base structure is maximally apparent to the user. Standard algorithms do not solve this problem (Eades and Tamassia, 1989). In our prototype we developed and implemented a number of algorithms. These algorithms are invoked by the user until the most appropriate layout is achieved. One of the algorithms based on a minimum distance and prediction technique is outlined below.

When a layout has been determined, a fisheye viewing technique is used to view the structure. A fisheye view of a graphical structure displays an area of interest around a focal point in detail and the rest of the graph in lesser detail with distance from the focal point (Furnas, 1986; Sarker and Brown, 1992). Such a view provides local details within the context of the overall structure. The manner in which it is applied here is somewhat similar to that proposed by Sarker & Brown. However, rather than utilize *a priori importance* of objects, a *level of importance* (*LI*) approach is taken in which objects are assigned to categories of importance at the time of construction of the knowledge base. These level of importance values are used in determining the *visual worth* of objects. Only those objects that have a visual worth greater than a criterion are displayed. The visual worth for any object increases as the of the level of importance object increases and decreases as distance from the focal point increases.

Initial Layout

For the initial layout nodes are positioned such that they do not overlap. The size of the nodes varies according to their LI value. The minimum distance, d_{min} , between nodes is that which is required to prevent overlap. If two nodes are connected by an arc the minimum distance is simply d_{min}. If two nodes i and j are not connected, then the minimum distance is

$$d_{min} * (1.2)^{n-1}$$

where n is the number of relations of node j. The algorithm for the initial layout is:

begin for LI = 1 down to 0 loop generate random position for object(i) loop until position is accepted generate random position for object (i+1) check the distance from this object to other objects if the object is far enough from all position is accepted else position is not accepted endloop endloop end.

Following application of this algorithm, nodes will not overlap, and nodes with greater number of relations will be further apart.

Optimizing the Layout

In the prototype a number of methods have been developed for optimizing the layout. One method will be briefly described here. This method, which uses the approach of minimizing distance between related nodes, moves the nodes incrementally closer to each other ensuring that they do not overlap. In the event that nodes will overlap it predicts a more suitable location and continues. This movement of nodes is shown in Fig 1. If a node has two or more relations it moves along the resultant vector.

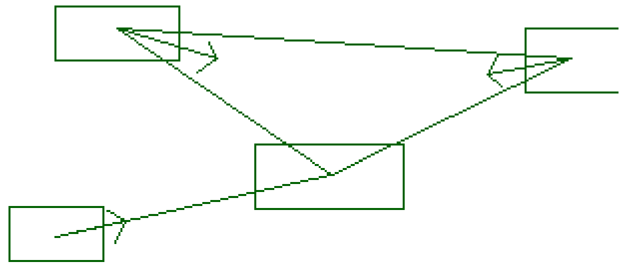

Fig 1 : Movement of the nodes

According as the nodes are moving closer to each other, the algorithm checks the distance to other objects. If some other node is less than the minimum distance, then using our prediction technique a new position is generated and the node jumps to this better location.

In Fig 2 for example, the node at position x_2,y_2 is moving towards the node at x_5,y_5. Rather than overlap the node at x_4,y_4 a new position will be predicted when it gets too close. The node will then jump to this position away from x_4,y_4.

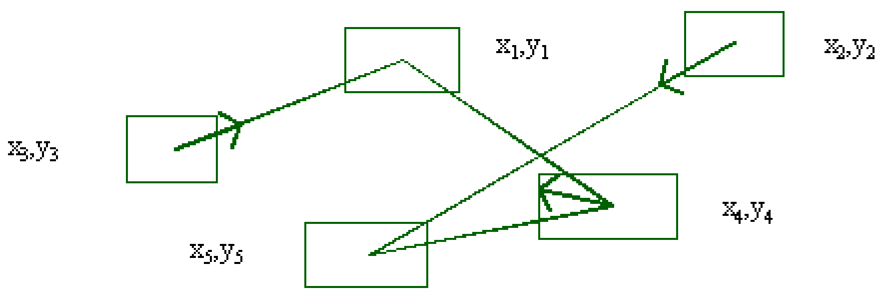

Fig 2: Node movement and prediction

In another of the optimization methods that has been developed, the new position of an object is defined by the centroid heuristic as the weighted mean of the positions of all related objects. SMP objects can be related by more than one type of relation, so each position is weighted by the number of arcs that directly connect the nodes. The weighting causes an object to move closer to the objects with which it shares the most relationships. If the computed centroid is less than the minimum distance specified then a new position is determined along a path between the original position and the centroid.

The prototype uses the various algorithms, under user control, to come up with the best layout. When the user is happy that this has been achieved the fisheye transformations are applied.

Conclusion

Comprehending the structure of a large knowledge base is a quite difficult cognitive task. A graphical representation can be of great assistance in exploring and understanding the structure of a SMP model. The layout of a large knowledge base into a graphical structure is a difficult problem and the method outlined here has produced very satisfactory results.

In particular the algorithm we developed for optimising the layout using a prediction method has been very successful.

References

Eades, P. and Tamassia, R. (1989) "Algorithms for Drawing Graphs: An Annotated Bibliography", Technical Report CS-89-90, Department of Computer Science, Brown University, Providence, RI

Furnas, G.W. (1986), "Generalized Fisheye Views", Proceedings CHI '86 Human Factors in Computing Systems, pp 16-23, New York

Mackinlay, J.D., Robertson, G.G., and Card, S.K. (1991), "The Perspective Wall: Detail and Context Smoothly Integrated", Proceedings ACM SIGCHI '91 Conference on Human Factors in Computing Systems, pp 173-179.

Robertson, G.G., Mackinlay, J.D. and Card, S.K. (1991), "Cone trees: Animated 3D Visualizations of Hierarchical Information", Proceedings ACM SIGCHI '91 Conference on Human Factors in Computing Systems, pp 189-194

Sarker, M. and Brown, M.H. (1992), "Graphical Fisheye Views of Graphs", Technical Report, Systems Research Center, Digital, Palo Alto, California