DYNAMIC WORLD SIMULATION FOR PLANNING WITH MULTIPLE AGENTS

Sharon Wood
Cognitive Studies Programme
University of Sussex,
Falmer, Brighton

ABSTRACT

The background to this work is the simulation of a planner who operates in a dynamic environment, by recognising the plans of others and predicting future events. The domain is that of a driver who must negotiate a route, obeying traffic regulations, and taking account of and avoiding collisions with other vehicles- The world of the driver is represented by a changing world model; the channel of communication between planner and this world model *are* actions from planner (driver's mind) to world, and perceptions from world to planner. Simulating a dynamic world model has been an important aspect of this work, and it is the implementation of this which is discussed below. Some new techniques of simulating physical processes have been developed.

I SCENARIO

The activities of the driver are modelled by a planning system, AUTODRIVE, written in PROLOG. The simulation of planning is for a single driver; however, AUTODRIVE can be consulted for any number of drivers, so in effect it is a planning system for multiple agents. The planner takes into account other drivers in the domain when constructing plans, as well as road signs, traffic signals, lane markings, etc., as these apply. For such purposes the driving domain is well-defined (confined to behaviour which occurs on the road and independent of the actors taking part - i.e. their personalities). This owes much to it being a relatively explicitly rule-bound domain, with 'rights of way' specifying legal sequences of movements between actors, and traffic laws governing, for instance, when we are permitted to overtake.

The aim of the planner is to construct a sequence of actions which enable the driver to reach a destination, following a specified route. The driver's movements (actions) are communicated to the world simulator as changes in location over a series of points in time. For instance:

```
t    [green_car,lewes_road,1.4m,north,0.5m]
t+1  [green_car,lewes_road,2.1m,north,0.5m]
t+2  [green_car,lewes_road,2.8m,north,0.5m]
t+3  [green_car,lewes_road,3.5m,north,0.5m]
t+4  [green_car,lewes_road,4.2m,north,0.5m]
```

This represents a green car travelling north along the Lewes Road (Brighton!), one half metre from the kerb, at a rate of 0.7 metres per unit of time.

The planning system achieves its goal - the driver reaching his destination - by plotting out in this way the series of moves the driver takes. These moves must, of course, be legal and not result in collisions with other cars or the violation of traffic regulations.

The simulated world of the driver is composed of roads whose width is divided into one or more lanes, which in places are marked, for example, with give-way lines, pedestrian crossings, and aouble yellow lines. Along the roadside are traffic signals and various signs, e.g. school ahead, no-entry, no right turn. Occasionally there are crossroads, roundabouts, side-turnings and T-junctions. The world is also populated with other vehicles moving freely about, and pedestrians (on crossings only!). These objects are selected as those most directly relevant to the task of driving.

At each instant of time, the world simulator, also written in PROLOG provides the driver with information about the world appropriate to his location and the direction his car is facing in - rather like a snapshot of the world. For instance:

```
newdata(green_car,[lines,yellow,double,10])
newdata(green_car,[lines,centre,broken,0])
newdata(green_car,[lanes,north,left,6,0])
newdata(green_car,[lanes,south,left,6,0])
newdata(green_car,[sign,no_entry,9,17])
newdata(green_car,
     [junction,left,inverness_rd,25,30])
newdata(green_car,
     [junction,right,aberdeen_rd,44,51])
newdata(green_car,
     [junction,right,edinburgh_rd,9,17])
```

This data is expressed in relation to the observer (green car), i.e. there are double yellow lines 10 metres ahead of the driver; a single left lane (6 metres wide) in the driver's direction and in the opposite direction, divided by a broken white line; there are three junctions ahead: on the left Inverness Road, from 25 to 30 metres; and on the right Aberdeen Road, 44 to 51 metres from the driver; and Edinburgh Road, 9 to 17 metres away.

Data about other vehicles is expressed independently of the observer:

```
newdata(green_car,[oncoming,
     [red_car,lewes_road,9.7m,south,0.5m]])
newdata(green_car,[in_left_lane,
     [red_car,lewes_road,9.7m,south,0.5m]])
newdata(green_car,[ahead,
     [blue_car,lewes_road,10.6m,north,0.5m]])
newdata(green_car,[in_left_lane,
     [blue_car,lewes_road,10.6m,north,0.5m]])
newdata(green_car,[behind,
     [yellow_car,lewes_road,0.5m,north,0.5m]])
newdata(green_car,[in_left_lane,
     [yellow_car,lewes_road,0.5m,north,0.5m]])
```

This tells the driver of the green car there is an oncoming (red) car in the left lane of the south carriageway; a (blue) car ahead; and a (yellow) car behind.

Feedback is also given on the driver's own position:

```
newdata(green_car,[you_are_here,
     [green_car,lewes_road,4.2m,north,0.5m]])
newdata(green_car,[in_left_lane,
     [green_car,lewes_road,4.2m,north,0.5m]])
```

From this range of information, the planner will attend to that which is relevant to the plan under construction; that which is novel, and that which violates expectations.

II MODELLING DYNAMIC ENVIRONMENTS

The term 'dynamic' world model is used here in contrast with an environment where no changes occur independently of the planner, only those which a single agent brings about himself. With an increase in number of agents who bring about changes in the environment, or the occurrence of independent events (e.g. rain), the environment is 'dynamic'.

The simulated world model plays a vital role in planning in such an environment. In the world of the single agent, actions may be reflected by 'instantaneous' additions and deletions of facts from the description of the world. However, in a dynamic environment, the planner cannot depend on knowledge of his own actions and the changes they bring about in the world, to successfully update his world model. Even STRIPS which planned in the world of a single agent, SHAKEY the robot, needed PLANEX (Fikes, Hart & Nilsson, 1972) to provide feedback on the success of its actions. Events and actions by other agents bring about changes in the world, rendering some plans ineffective; the planner therefore requires a constant source of up-to-date information, which an independent simulation of the world can provide.

Relating the actions of a number of agents through the medium of time is a major problem in providing an accurate world description or simulation of the environment. The representation of time becomes a *Very* important issue. Work in this area, however, is not advanced. Proposed systems fall into two categories: (a) updating through predicted significant events; (b) updating incrementally through some pre-specified unit of time. Hendrix (1973) and Birtwhistle et al (1979) use the first approach, while AUT0DR1VE uses an alternative approach, similar to Wesson (1977) based on the second.

The unitisation of time in the world simulation, and thus the output of time-related observational data, is fundatmental to the planning process. For from over a series of time-related observations, a vehicle's speed and direction, and their rate of change, can be deduced. From these observations the planning system derives (through hypothesis) both another agent's intention, and a projection of his movement or actions in achieving this goal. This constitutes the planning system's "plan lookahead" model on which it may base future actions.

III DESCRIPTION OF THE SIMULATION SYSTEM

The data for representing the driving environment is obtained from map information and observations of real roads. This forms a database of descriptions of task specific road elements, such as road markings, signs, etc. A driver is capable of covering quite a large distance in a short space of time, therefore the data which is potentially available is divided into manageable 'chunks'. To effect this, road and junction areas are divided into 'segments' and treated as separate packages of data; where only several segments at a time constitute the world model of the driver. The descriptions of objects are linked to the segment in which they appear, stating their precise location within it.

The location of the driver is passed to the world simulator. The data item associated with the car - its 'car descriptor' - informs the simulator of which segment the driver is in, its position within that segment, its distance from the kerb, and the direction in which it is travelling.

For each segment there is a 'segment controller' whose takes a vehicle's location and decides whether it is close enough to receive information about that segment. The segment controller plays the part of intermediary, and so does not itself obtain the data for the driver. Instead, the segment controller in effect 'plots' the location of the driver by informing each item in its segment of that location; each item then 'shouts out' when it thinks it can 'see' the vehicle (and therefore the driver can see it!). Thus, although the simulated world model represents a dynamic environment, it is itself static, where vehicles are 'moving objects' within that world, whose locations are plotted for each snapshot of time.

Each car within the driving environment is also an observable object; therefore, some code is created for the driver, to enable it to 'shout out[1] about itself, as other objects do. The lane position of the car is made available by the same mechanism.

The program incorporating these techniques so far described is implemented and running in PROLOG. Extensions are proposed: The simulator will ensure that all events taking place operate within the physical constraints imposed by the environment. Thus although the planner intends to be in a certain location at a particular point in time, the world simulator interprets actions in accordance with such constraints.

Creating the car as an identifiable object also proviaes a 'vehicle' for the feedback mechanism to the driver on his own position (i.e. "you are here . . ."), in accordance with this, allowing the planner to maintain an accurate representation of the world. For instance, a car may pick up speed going down hill; the driver is able to perceive this via the feedback he receives.

Similarly, dynamic world simlation also involves modelling interactions between actions and events and the objects within that world. Problems surround attempting to model accurately the side-effects of events and actions by agents in a representation of the world. In earlier problem solving work (e.g. Fikes, Hart and Nilsson, 1972) it was relatively easy to ascertain the side-effects of actions by a single agent when the world itself was unchanging. For instance, a change in location meant the agent was no longer in place A and was now in place B. One could also make inferences: for instance, if object X was also in place B, the agent was next to object X, etc.

A changing environment may affect our ability to achieve our goals by no longer satisfying necessary preconditions. A more subtle effect of a changing environment, however, is to allow actions to take place but change their resulting side-effects, i.e. what they actually achieve. For example, one may estimate that driving at a certain speed would enable one to brake within a certain distance; however, if whilst executing this action, some events took place - e.g. some oil spilt onto the road, the effect of braking will be different; i.e. the car will travel a lot further before coming to a halt.

This method of simulation offers new scope for handling this problem. The world simulator mediates the actions of agents, rather than allowing them to directly change the world model. Consequently, the simulator is able to constrain those actions; for instance, the current velocity of a vehicle will determine the minimum distance it can possibly cover in a specified time, even at maximum deceleration. Further, the world simulator is also in a position to model indirect side-effects of events in the world, which consequently reflects on the quality of information available to each driver. For instance, it is not part of the act of driving down a road that a large lorry or bus is actually making any changes to what there is to see. However, it may well be the case that a side-effect of a small car being close behind a bus, means that objects it would otherwise be able to see are obscured. So although a lorry driver does not himself effect any such changes in the world, these side-effects must be reflected in the quality of information available to the driver of the car.

From this point of view, segmentation is a powerful idea. By monitoring all that goes on within a fixed *area* or segment, segment controllers are in a position to monitor for interactions between agents, and between their actions and various states of the world. They are then able to model this by exercising control over what information gets through to the planner.

IV <u>CONCLUSION</u>

A world simulator for modelling a dynamic environment has been described. The method of simulation is based on the principle that it is easier to operate with a fixed representation of part of the dynamic world, than with a representation whose defining characteristics *are* also changing.

Some techniques of modelling physical processes have been developed under the general heading of segment control. Because the process of simulation resides with individual objects over which the segment controller presides, monitoring for interactions between agents and objects is localised and can be reflected in the output allowed by the segment controller.

Segment controllers could potentially be quite sophisticated, handling the effects of conditions such as fog on the output, e.g. all items at a distance greater than known visibility *are* vetoed. Hopefully, the performance of the planner would degrade gracefully with the diminishing quality of information (just like human planners), rather than merely grinding to a halt.

ACKNOWLEDGEMENTS

The author wishes to thank Dr Aaron Sloman and Dr Steven Hardy for their comments and advice during this research and in the preparation of this paper.

REFERENCES

[1] BIRTWHISTLE, G., <u>DEMOS</u>: A <u>System for Discrete Event Modelling on Simula</u>, Macmillan, London, 1976

[2] FIKES, R.E., HART, P.E. & NILSSON, N.J., 'Some New Directions in Robot Problem Solving', <u>Machine Intelligence</u>, 3, 1972

[3] HENDRIX, G.G., 'Modelling Simultaneous Actions and Continuous Processes', <u>Artificial Intelligence</u>, 4, 1973, pp145-180

[4] WESSON, R.B., 'Planning in the World of the Air Traffic Controller', <u>International Joint Conference on Artificial Intelligence</u>, 5, 1977